(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
18 November 2004 (18.11.2004)

PCT

(10) International Publication Number
WO 2004/100043 A1

(51) International Patent Classification⁷: G06F 17/60

(21) International Application Number: PCT/US2004/008626

(22) International Filing Date: 22 March 2004 (22.03.2004)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
10/425,610 30 April 2003 (30.04.2003) US

(71) Applicant (*for all designated States except US*): **GE FINANCIAL ASSURANCE HOLDINGS, INC.** [US/US]; 6604 West Broad Street, Richmond, VI 23230 (US).

(72) Inventors; and
(75) Inventors/Applicants (*for US only*): **BONISSONE, Piero, Patrone** [IT/US]; 1065 Avon Road, Schenectady, NY 12308 (US). **SUBBU, Rajesh, Venkat** [IN/US]; 65, 25th Street, Troy, NY 12180 (US). **YAN, Weizhong** [US/US]; 3 Hampshire Place, Clifton Park, NY 12065 (US). **CHAKRABORTY, Anindya** [IN/US]; 1A4 Sheridan Village, Schenectady, NY 12308 (US).

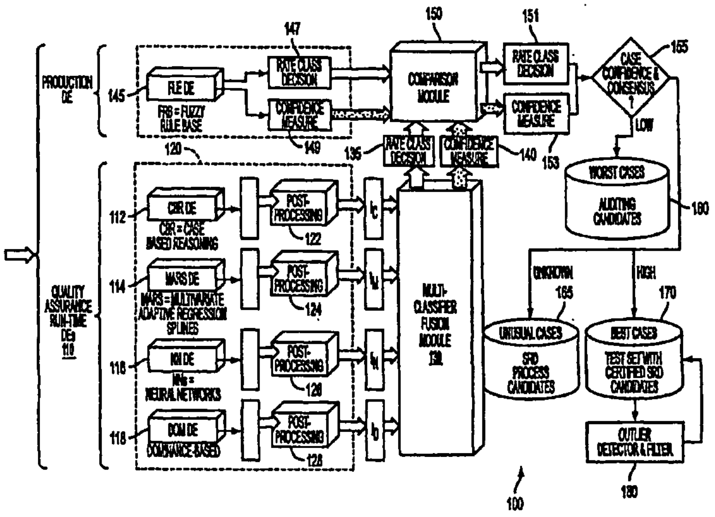
(74) Agents: **HAYDEN, Scott** et al.; Patent Counsel, General Electric Company, 3135 Easton Turnpike (W3C), Fairfield, CT 06828 (US).

(81) Designated States (*unless otherwise indicated, for every kind of national protection available*): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

(84) Designated States (*unless otherwise indicated, for every kind of regional protection available*): ARIPO (BW, GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

[Continued on next page]

(54) Title: SYSTEM AND PROCESS FOR A NEURAL NETWORK CLASSIFICATION FOR INSURANCE UNDERWRITING SUITABLE FOR USE BY AN AUTOMATED SYSTEM



(57) **Abstract:** A method and system for designing a neural network classifier (116) and such a neural network for an automated insurance underwriting system and/or its quality assurance is described. While the design method is demonstrated for quality assurance of automated insurance underwriting (100), it is broadly applicable to diverse decision-making applications in business, commercial, and manufacturing processes. Specifically, multi-class classification problems are solved by decomposing a multi-class classifier into multiple binary-classifiers (5010, 5020, 5030), which reduces the complexity of the neural network structure, thus reducing the training time and improving the classification performance. Furthermore, the invention also describes a method to incorporate the domain knowledge into the neural network classifier. Both methods work to improve the performance of the classifier.

WO 2004/100043 A1

Declarations under Rule 4.17:

- *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii)) for the following designations AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, UZ, VC, VN, YU, ZA, ZM, ZW, ARIPO patent (BW, GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG)*
- *as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii)) for the following designations AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, UZ, VC, VN, YU, ZA, ZM, ZW, ARIPO patent (BW, GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG)*

Published:

- *with international search report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

SYSTEM AND PROCESS FOR A NEURAL NETWORK CLASSIFICATION FOR INSURANCE UNDERWRITING SUITABLE FOR USE BY AN AUTOMATED SYSTEM

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to a system and process for underwriting insurance applications, and more particularly to a system and process for underwriting insurance applications based on a neural network classification.

2. Description of Related Art

Classification is the process of assigning an input pattern to one of a predefined set of classes. Classification problems exist in many real-world applications, such as medical diagnosis, machine fault diagnosis, handwriting character recognition, fingerprint recognition, and credit scoring, to name a few. Broadly speaking, classification problems can be categorized into two types: dichotomous classification, and polychotomous classification. Dichotomous classification deals with two-class classification problems, while polychotomous classification deals with classification problems that have more than two classes.

Classification consists of developing a functional relationship between the input features and the target classes. Accurately estimating such a relationship is key to the success of a classifier. Insurance underwriting is one of these classification problems. The underwriting process consists of assigning a given insurance application, described by its medical and demographic records, to one of the risk categories (also referred to as rate classes). A trained individual or individuals traditionally perform insurance underwriting. A given application for insurance (also referred to as an "insurance application") may be compared against a plurality of underwriting standards set by an insurance company. The insurance application may be classified into one of a plurality of risk categories available for a type of insurance coverage requested by an applicant. The risk categories then affect the premium paid by the applicant, *e.g.*, the higher the risk category, higher the premium. A decision to accept or reject the application for insurance may also be part of this risk classification, as risks above a certain tolerance level set by the insurance company may simply be rejected.

Insurance underwriting often involves the use of a large number of features in the decision-making process. The features typically include the physical conditions, medical information, and family history of the applicant. Further, insurance underwriting frequently has large number of risk categories (rate classes). The risk category of an insurance application is traditionally determined by using a number of rules/standards, which have the form of, for example, "if the value of feature *x* exceeds *a*, then the application can't be rate class *C*, *i.e.*, the application has to be lower than *C*". Such manual underwriting, however, is not only time-consuming, but also often inadequate in consistency and reliability. The inadequacy becomes more apparent as the complexity of insurance applications increases.

There can be a large amount of variability in the insurance underwriting process when individual underwriters perform it. Typically, underwriting standards cannot cover all possible cases and variations of an application for insurance. The underwriting standards may even be self-contradictory or ambiguous, leading to an uncertain application of the standards. The subjective judgment of the underwriter will almost always play a role in the process. Variation in factors such as underwriter training and experience, and a multitude of other effects can cause different underwriters to issue different, inconsistent decisions. Sometimes these decisions can be in disagreement with the established underwriting standards of the insurance company, while sometimes they can fall into a "gray area" not explicitly covered by the underwriting standards.

Further, there may be an occasion in which an underwriter's decision could still be considered correct, even if it disagrees with the written underwriting standards. This situation can be caused when the underwriter uses his/her own experience to determine whether the underwriting standards should be adjusted. Different underwriters may make different determinations about when these adjustments are allowed, as they might apply stricter or more liberal interpretations of the underwriting standards. Thus, the judgment of experienced underwriters may be in conflict with the desire to consistently apply the underwriting standards.

Other drawbacks may also exist.

SUMMARY OF THE INVENTION

According to an exemplary embodiment of the invention, a system for underwriting an insurance application based on a plurality of previous insurance application underwriting decisions, where underwriting the insurance application includes assigning a classification to the insurance application, is described. The system includes a processing module, where the processing module processes a plurality of features associated with the insurance application and outputs a processed result for each of the plurality of features, a plurality of classifiers, where there is a classifier for each classification and where each of the classifiers receives the plurality of processed results and outputs a classification result for the insurance application, at least one post-processor module, where the at least one post-processing module processes each of the classification results and outputs a processed classification result, and a classification selector for selecting one of the processed classification results, where the selection indicates a classification assignment for the insurance application.

By way of a further exemplary embodiment of the invention, a system for underwriting an insurance application based on a plurality of previous insurance application underwriting decisions, where underwriting the insurance application includes assigning a classification to the insurance application, includes a processing module, where the processing module processes a plurality of features associated with the insurance application, where the processing performed by the pre-processing module comprises at least one of range normalization and generating a tag, wherein the tag is generated based on one or more rules applied to at least one of the plurality of features, outputs a processed result for each of the plurality of features, a plurality of binary classifiers, where there is a binary classifier for each classification and where each of the binary classifiers receives the plurality of processed results and outputs a classification result for the insurance application, at least one post-processor module, where the at least one post-processing module processes each of the classification results and outputs a processed classification result, and a classification selector for selecting one of the processed classification results, where the selection indicates a classification assignment for the insurance application and selects the highest of the processed classification results.

According to another exemplary embodiment, a process for underwriting an insurance application based on a plurality of previous insurance application underwriting decisions, where underwriting the insurance application includes assigning a classification to the insurance application, includes receiving the insurance application, where the insurance application has a plurality of features, processing a plurality of features associated with the insurance application, generating a processed result for each of the plurality of features, generating a classification result for the insurance application based on the processed result, processing the classification results, generating a processed classification result, and selecting one of the processed classification results, where the selection indicates a classification assignment for the insurance application.

According to a further embodiment of the invention, a process for underwriting an insurance application based on a plurality of previous insurance application underwriting decisions, where underwriting the insurance application includes assigning a classification to the insurance application, includes receiving the insurance application, where the insurance application has a plurality of features, processing a plurality of features associated with the insurance application, where processing a plurality of features further comprises at least one of range normalization and generating a tag, wherein the tag is generated based on one or more rules applied to at least one of the plurality of features, and generating a processed result for each of the plurality of features. The process further includes generating a classification result for the insurance application based on the processed result, where the classification results are generated by a plurality of binary classifiers, and where there is a classifier for each classification, processing the classification results, generating a processed classification result, and selecting one of the processed classification results, where the selection indicates a classification assignment for the insurance application and selects the highest of the processed classification results.

By way of a further exemplary embodiment of the invention, a system for underwriting an insurance application based on a plurality of previous insurance application underwriting decisions, where underwriting the insurance application includes assigning a classification to the insurance application, is described. The system includes means for receiving the insurance application, where the insurance application has a plurality of features, means for processing a plurality of features associated with the insurance application, means for generating a processed result for each of the plurality of features, means for generating a classification result for the insurance application based on the processed result, means for processing the classification results, means for generating a processed classification result, and means for selecting one of the processed classification results, where the selection indicates a classification assignment for the insurance application.

According to another example of an embodiment of the invention, a system for underwriting an insurance application based on a plurality of previous insurance application underwriting decisions, where underwriting the insurance application includes assigning a classification to the insurance application, includes means for receiving the insurance application, where the insurance application has a plurality of features, means for processing a plurality of features associated with the insurance application, where processing a plurality of features further comprises at least one of range normalization and generating a tag, wherein the tag is generated based on one or more rules applied to at least one of the plurality of features, means for generating a processed result for each of the plurality of features, means for generating a classification result for the insurance application based on the processed result, where the classification results are generated by a plurality of binary classifiers, and where there is a classifier for each classification, means for processing the classification results, means for generating a processed classification result and means for selecting one of the processed classification results, where the selection indicates a classification assignment for the insurance application and selects the highest of the processed classification results.

By way of a further exemplary embodiment of the invention, a computer readable medium having code for causing a processor to underwrite an insurance application based on a plurality of previous insurance application underwriting decisions, where underwriting the insurance application includes assigning a classification to the insurance application, includes code for receiving the insurance application, where the insurance application has a plurality of features, code for processing a plurality of features associated with the insurance application, code for generating a processed result for each of the plurality of features, code for generating a classification result for the insurance application based on the processed result, code for processing the classification results, code for generating a processed classification result, and code for selecting one of the processed classification results, where the selection indicates a classification assignment for the insurance application.

By way of a further exemplary embodiment of the invention, a computer medium having code for causing a processor to underwrite an insurance application based on a plurality of previous insurance application underwriting decisions, where underwriting the insurance application includes assigning a classification to the insurance application, includes code for receiving the insurance application, where the insurance application has a plurality of features, code for processing a plurality of features associated with the insurance application, where processing a plurality of features further comprises at least one of range normalization and generating a tag, wherein the tag is generated based on one or more rules applied to at least one of the plurality of features, code for generating a processed result for each of the plurality of features, code for generating a classification result for the insurance application based on the processed result, where the classification results are generated by a plurality of binary classifiers, and where there is a classifier for each classification, code for processing the classification results, code for generating a processed classification result, and code for selecting one of the processed classification results, where the selection indicates a classification assignment for the insurance application and selects the highest of the processed classification results.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 illustrates the architecture of a quality assurance system based on the fusion of multiple classifiers according to an embodiment of the invention.

Figure 2 illustrates a table of an outer product using the function T(x,y) according to an embodiment of the invention.

Figure 3 illustrates the disjointed rate classes within the universe of rate classes according to an embodiment of the invention.

Figure 4 illustrates the results of the intersections of the rate classes and the universe according to an embodiment of the invention.

Figures 5-9 illustrate the results of T-norm operators according to an embodiment of the invention.

Figures 10-14 illustrate the normalized results of T-norm operators according to an embodiment of the invention.

Figure 15 illustrates a summary of the fusion of two classifiers according to an embodiment of the invention.

Figure 16 illustrates a penalty matrix for a fusion module according to an embodiment of the invention.

Figure 17 illustrates a summary of the fusion of two classifiers with disagreement according to an embodiment of the invention.

Figure 18 illustrates a summary of the fusion of two classifiers with agreement and discounting according to an embodiment of the invention.

Figures 19-23 illustrate the results of T-norm operators according to an embodiment of the invention.

Figures 24-28 illustrate the normalized results of T-norm operators according to an embodiment of the invention.

Figure 29 illustrates a Dempster-Schaefer penalty matrix according to an embodiment of the invention.

Figure 30 illustrates a comparison matrix according to an embodiment of the invention.

Figure 31 illustrates fusion as a function of a confidence threshold for non-nicotine cases according to an embodiment of the invention.

Figure 32 illustrates fusion as a function of a confidence threshold for nicotine cases according to an embodiment of the invention.

Figure 33 illustrates a Venn diagram for fusion for non-nicotine cases according to an embodiment of the invention.

Figure 34 illustrates a Venn diagram for fusion for nicotine cases according to an embodiment of the invention.

Figure 35 is a flowchart that illustrates an outlier detector according to an embodiment of the invention.

Figure 36 illustrates an outlier detector used in quality assurance according to an embodiment of the invention.

Figure 37 illustrates a plot of two features for insurance applications according to an embodiment of the invention.

Figure 38 is a flowchart that illustrates a tuning process according to an embodiment of the invention.

Figure 39 is a flowchart that illustrates a classification process according to an embodiment of the invention.

Figure 40 illustrates a comparison matrix according to an embodiment of the invention.

Figure 41 illustrates a comparison matrix for a modified process according to an embodiment of the invention.

Figure 42 is a flowchart that illustrates a multi-variate adaptive regression splines ("MARS") process according to an embodiment of the invention.

Figure 43 is a histogram that illustrates decision boundaries according to an embodiment of the invention.

Figure 44 illustrates a parallel network implementation according to an embodiment of the invention.

Figure 45 illustrates a comparison matrix according to an embodiment of the invention.

Figure 46 illustrates an annotated comparison matrix according to an embodiment of the invention.

Figure 47 illustrates a performance of MARS models using five partitions according to an embodiment of the invention.

Figure 48 illustrates minimum, maximum, and average performances of a network of MARS models according to an embodiment of the invention.

Figure 49 illustrates a piecewise-continuous classification boundary in a feature space according to an embodiment of the invention.

Figure 50 illustrates a multi-class neural network decomposed into multiple binary classifiers according to an embodiment of the invention.

Figure 51 illustrates an architecture for a neural network classifier according to an embodiment of the invention.

Figure 52 illustrates a confusion matrix before post-processing according to an embodiment of the invention.

Figure 53 illustrates a confusion matrix after post-processing according to an embodiment of the invention.

Figure 54 illustrates performance before post-processing according to an embodiment of the invention.

Figure 55 illustrates performance after post-processing according to an embodiment of the invention.

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

A system and process for underwriting of insurance applications that is suitable for use by a computer rather than by human intervention is described. The system and process make use of existing risk assignments made by human underwriters to categorize new applications in terms of the risk involved. One technical effect of the invention is to provide an automated process for consistent and accurate underwriting decisions for insurance applications. Various aspects and components of this system and process are described below.

It will be recognized, however, that the principles disclosed herein may extend beyond the realm of insurance underwriting and that it may be applied to any risk classification process, of which the determination of the proper premium to cover a given risk (*i.e.* insurance underwriting) is just an example. Therefore the ultimate domain of this invention may be considered risk classification.

1. Fusion Module

An aspect of the invention provides a system and process for fusing a collection of classifiers used for an automated insurance underwriting system and/or its quality assurance. While the design method is demonstrated for quality assurance of automated insurance underwriting, it is broadly applicable to diverse decision-making applications in business, commercial, and manufacturing processes. A process of fusing the outputs of a collection of classifiers is provided. The fusion can compensate for the potential correlation among the classifiers. The reliability of each classifier can be represented by a static or dynamic discounting factor, which will reflect the expected accuracy of the classifier. A static discounting factor represents a prior expectation about the classifier's reliability, *e.g.*, it might be based on the average past accuracy of the model. A dynamic discounting represents a conditional assessment of the classifier's reliability, *e.g.*, whenever a classifier bases its output on an insufficient number of points, the result is not reliable. Hence, this factor could be determined from the post-processing stage in each model. The fusion of the data will typically result in some amount of consensus and some amount of conflict among the classifiers. The consensus will be measured and used to estimate a degree of confidence in the fused decisions.

According to an embodiment of the invention, a fusion module (also referred to as a fusion engine) combines the outputs of several decision engines (also referred to as classifiers or components of the fusion module) to determine the correct rate class for an insurance application. Using a fusion module with several decision engines may enable a classification to be assigned with a higher degree of confidence than is possible using any single model. According to an embodiment of the invention, a fusion module function may be part of a quality assurance ("QA") process to test and monitor a production decision engine ("PDE") that makes the rate class assignment in real-time. At periodic intervals, *e.g.*, every week, the fusion module and its components may review the decisions made by the PDE during the previous week. The output of this review will be an assessment of the PDE performance over that week, as well as the identification of cases with different level of decision quality.

The fusion module may permit the identification of the best cases of application classification, *e.g.*, those with high-confidence, high-consensus decisions. These best cases in turn may be likely candidates to be added to the set of test cases used to tune the PDE. Further, the fusion module may permit the identification of the worst cases of application classification, *e.g.*, those with low-confidence, low-consensus decisions. These worst cases may be likely candidates to be selected for a review by an auditing staff and/or by senior underwriters.

A fusion module may also permit the identification of unusual cases of application classification, *e.g.*, those with unknown confidence in their decisions, for which the models in the fusion module could not make any strong commitment or avoided the decision by routing the insurance application to a human underwriter. These cases may be candidates for a blind review by senior underwriters. In addition, a fusion module may also permit an assessment of the performance of the PDE, by monitoring the PDE accuracy and variability over time, such as monitoring the statistics of low, borderline and high quality cases as well as the occurrence of unusual cases. These statistics can be used as indicators for risk management.

According to an embodiment of the invention, a fusion module may leverage the fact that except for the unusual situation where all the components (*e.g.*, models) contain the same information (*e.g.*, an extreme case of positive correlation), each component should provide additional information. This information may either corroborate or refute the output of the other modules, thereby supporting either a measure of consensus, or a measure of conflict. These measures may define a confidence in the result of the fusion. In general, the fusion of the components' decisions may provide a more accurate assessment than the decision of each individual component.

The fusion module is described in relation to various types of decision engines, including a case-based decision engine, a dominance-based decision engine, a multivariate adaptive regression splines engine, and a neural network decision engine respectively. However, the fusion module may use any type of decision engine. According to an embodiment of the invention, the fusion module will support a quality assurance process for a production decision engine. However, it is understood that the fusion module could be used for a quality assurance process for any other decision making process, including a human underwriter.

According to an embodiment of the invention, a general method for the fusion process, which can be used with classifiers that may exhibit any kind of (positive, neutral, or negative) correlation with each other, may be based on the concept of triangular norms ("T-norm"), a multi-valued logic generalization of the Boolean intersection operator. The fusion of multiple decisions, produced by multiple sources, regarding objects (*e.g.*, classes) defined in a common framework (*e.g.*, the universe of discourse) consists of determining the underlying of degree of consensus for each object (*e.g.*, class) under consideration, *i.e.*, the intersections of their decisions. With the intersections of multiple decisions, possible correlation among the sources needs to be taken into account to avoid under-estimates or over-estimates. This is done by the proper selection of a T-norm operator.

According to an embodiment of the invention, each model is assumed to be solving the same classification problem. Therefore, the output of each classifier is a weight assignment that represents the degree to which a given class is selected. The set of all possible classes, referred to as U, represents the common universe of all answers that can be considered by the classifiers. The assignment of weights to this universe represents the classifier's ignorance (*i.e.*, lack of commitment to a specific decision). This is a discounting mechanism that can be used to represent the classifier's reliability.

According to an embodiment of the invention, the outputs of the classifiers may be combined by selecting the generalized intersection operator (*e.g.*, the T-norm) that better represents the possible correlation between the classifiers. With this operator, the assignments of the classifiers are intersected and a derived measure of consensus is computed. This fusion may be performed in an associative manner, *e.g.*, the output of the fusion of the first two classifiers is combined with the output of the third classifier, and so on, until all available classifiers have been considered. At this stage, the final output may be normalized (*e.g.*, showing the degree of selection as a percentage). Further, the strongest selection of the fusion may be identified and qualified with its associated degree of confidence.

Thus, according to an embodiment of the invention, a fusion module only considers weight assignments made either to disjoint subsets that contain a singleton (*e.g.*, a rate class) or to the entire universe of classes U (*e.g.*, the entire set of rate classes), as will be described in greater detail below. Once compensation has been made for correlation and fusion has been performed, the degree of confidence C is computed among the classifiers and used to qualify the decision obtained from the fusion. Further, the confidence measure and the agreement or disagreement of the fusion module's decision is used with the production engine's decision to assess the quality of the production engine. As a by-product, the application cases may be labeled in terms of the decision confidence. Thus, cases with low, high, or unknown confidence may be used in different ways to maintain and update the production engine.

Other types of aggregation could be used, but would need to be associative, compensate for correlation, accommodate the discounting of classifiers, and generate a confidence measure of the combined decision, properties that are not directly satisfied. A particular case may be a Dempster-Shafer ("DS") fusion rule. The DS fusion rule requires the classifiers to be evidentially independent, *i.e.,* the errors of one classifier must be uncorrelated with those of another one. Furthermore, the DS paradigm does not allow us to represent the ordering among the classes, typical of the insurance underwriting process. This ordering implies that there could be minor differences (such as the selection of two adjacent classes) and major differences (such as the selection of different classes at the extreme of their range). Therefore, the conflict between two sources is a gradual one, rather than a binary one (hit/miss). Finally, in DS theory, the classifiers' outputs are considered probability assignments.

Triangular norms (T-norms) and Triangular conorms (T-conorms) are the most general families of binary functions that satisfy the requirements of the conjunction and disjunction operators, respectively. T-norms T(x,y) and T-conorms S(x,y) are two-place functions that map the unit square into the unit interval, *i.e.*, T(x,y): [0,1]x[0,1] → [0,1] and S(x,y): [0,1]x[0,1] → [0,1]. They are monotonic, commutative and associative functions. Their corresponding boundary conditions, i.e., the evaluation of the T-norms and T-conorms at the extremes of the [0,1] interval, satisfy the truth tables of the logical AND and OR operators. They are related by the DeMorgan duality, which states that if N(x) is a negation operator, then the T-conorm S(x,y,) can be defined as S(x,y) = N(T(N(x), N(y))).

As described in Bonissone and Decker (1986) the contents of which are incorporated by reference in their entirety, six parameterized families of T-norms and their dual T-conorms may be used. Of the six parameterized families, one family was selected due to its complete coverage of the T-norm space and its numerical stability. This family has a parameter *p*. By selecting different values of *p*, T-norms with different properties can be instantiated, and thus may be used in the fusion of possibly correlated classifiers.

Various articles discuss the fusion and the different features associated therewith, include proofs as to the development of algorithms associated with the present invention. Chibelushi *et al.* (Chibelushi, C. C., Deravi, F., and Mason, J. S. D., "Adaptive Classifier Integration for Robust Pattern Recognition," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 29, no. 6, 1999, the contents of which are incorporated herein by reference) describe a linear combination method for combining the outputs of multiple classifiers used in speaker identification applications.

Fairhurst and Rahman (Fairhurst, M. C., and Rahman, A. F. R., "Enhancing consensus in multi expert decision fusion," *IEE Proc.-Vis. Image Signal Process*, vol. 147, no. 1, 2000, the contents of which are incorporated herein by reference) describe ENCORE, a multi-classifier fusion system for enhancing the performance of individual classifiers for pattern recognition tasks, specifically, the task of hand written digit recognition. Kuncheva and Jain (Kuncheva, L. I., and Jain, L. C., "Designing Classifier Fusion Systems by Genetic Algorithms," *IEEE Transactions on Evolutionary Computation,* vol. 4, no. 4, 2000, the contents of which are incorporated herein by reference) describe a genetic algorithm approach to the design of fusion of multiple classifiers.

Xu et al. (Xu, L., Krzyzak, A., and Suen, C. Y., "Methods of Combining Multiple Classifiers and Their Applications to Handwriting Recognition," *IEEE Transactions* *on Systems, Man, and Cybernetics*, vol. 22, no. 3, 1992, the contents of which are incorporated herein by reference) describe several standard approaches for classifier decision fusion, including the Dempster-Shafer approach, and demonstrate fusion for handwritten character recognition.

Arthur Dempster (A. P. Dempster, "Upper and lower probabilities induced by a multivalued mapping," *Annals of Mathematical Statistics*, 38:325--339, 1967, the contents of which are incorporated herein by reference) describes a calculus based on lower and upper probability bounds. Dempster's rule of combination describes the pooling of sources under the assumption of evidential independence. Glenn Shafer (G. Shafer, "*A Mathematical Theory of Evidence*", Princeton University Press, Princeton, New Jersey, 1976, the contents of which are incorporated herein by reference) describes the same calculus discovered by Dempster, but starting from a set of super-additive belief functions that are essentially lower bounds. Shafer derives the same rule of combination as Dempster. Enrique Ruspini (E. Ruspini, "Epistemic logic, probability, and the calculus of evidence. *Proc. Tenth Intern. Joint Conf. on Artificial Intelligence*, Milan, Italy, 1987, the contents of which are incorporated herein by reference) goes on to describe a possible-world semantics for Dempster-Shafer theory.

B. Schweizer and A. Sklar (B. Schweizer and A. Sklar, "Associative Functions and Abstract Semi-Groups", *Publicationes Mathematicae Debrecen*, 10:69-81, 1963, the contents of which are incorporated herein by reference) describe a parametric family of triangular T-norm functions that generalize the concept of intersection in multiple-valued logics. Piero Bonissone and Keith. Decker (P. P. Bonissone and K. Decker, "Selecting Uncertainty Calculi and Granularity: An Experiment in Trading-off Precision and Complexity" in Kanal and Lemmer (editors*) Uncertainty in Artificial Intelligence*, pages 217-247, North-Holland, 1986, the contents of which are incorporated herein by reference) describe an experiment based on Schweizer and Sklar's parameterized T-norms. They show how five triangular norms can be used to represent an infinite number of t-norm for some practical values of information granularity. Piero Bonissone (P. P. Bonissone, "Summarizing and Propagating

Uncertain Information with Triangular Norms", *International Journal of Approximate Reasoning,* 1(1):71-101, January 1987, the contents of which are incorporated herein by reference) also describes the use of Triangular norms in dealing with uncertainty in expert system, Specifically he shows the use Triangular norms to aggregate the uncertainty in the left-hand side of production rules and to propagate it through the firing and chaining of production rules.

Fig. 1 illustrates the architecture of a quality assurance system based on the fusion of multiple classifiers according to an embodiment of the invention. These classifiers may include case-based reasoning model (described in U.S. Patent Application Serial Nos. 10/170,471 and 10/171,190, the contents of which are incorporated herein by reference), a multivariate adaptive regression splines model (hereinafter also referred to as "MARS"), a neural network model and a dominance-based model. The MARS, neural networks, and dominance-based models are all described in greater detail below.

System 100, as illustrated in Fig. 1, includes a number of quality assurance decision engines 110. In the embodiment illustrated in Fig. 1, the quality assurance decision engines 110 comprise a case-based reasoning decision engine 112, a MARS decision engine 114, a neural network decision engine 116, and a dominance-based decision engine 118. It is understood, however, that other types of quality assurance decision engines 110 could be used in addition to and/or as substitutes for those listed in the embodiment of the invention illustrated in Fig. 1.

Post processing modules 122, 124, 126, and 128 receive the outputs from the various quality assurance decision engines 120 and perform processing on the outputs. The results of the post-processing are input into a multi-classifier fusion module 130. The multi-classifier fusion module 130 then outputs a fusion rate class decision 135 and a fusion confidence measure 140, which are input into comparison module 150.

A fuzzy logic rule-based production engine 145 outputs a production rate class decision 147 and a production confidence measure 149, which are then input into comparison module 150. After a comparison has been made between the production rate class decision 147 and the fusion rate class decision 135, and the production confidence measure 149 and the fusion confidence measure 140, a compared rate class decision 151 and a compared confidence measure 153 are output by comparison module 150. An evaluation module 155 evaluates the case confidence and consensus regarding the compared rate class 151 and the compared confidence measure 153. Those cases evaluated as "worst cases" are stored in case database 160, and may be candidates for auditing. Those cases evaluated as "unusual cases" are stored in case database 165, and may be candidates for standard underwriting. Those cases evaluated as "best cases" are stored in case database 170, and may be candidates for using with the test sets. The outlier detector and filter 180 may ensure that any new addition to the best-case database 170 will be consistent (in the dominance sense described below) with the existing cases, preventing logical outliers from being used. System 100 of Fig. 1 will now be described in greater detail below.

According to an embodiment of the invention, the fusion process as disclosed in Fig. 1 includes four general steps. These steps are: (1) collection, discounting and post-processing of modules' outputs; (2) determination of a combined decision via the associative fusion of the modules' outputs; (3) determination of degree of confidence; and (4) identification of cases that are candidates for test set, auditing, or standard reference decision process, via the comparison module 150. These steps will now be described in greater detail below.

Each quality assurance decision module 110 generates an output vector $\boldsymbol{I} = [I(1), I(2), \ldots I(N+1)]$ where $I(i) \in [0,M]$, where M is a large real value and N is the number of rate classes. In the embodiment of the invention illustrated in Fig. 1, each vector $\boldsymbol{I}$ is identified by a superscript associated with the quality assurance decision module 120 that generates the vector. Therefore, I^C is generated by case-based reasoning decision engine 112, I^M is generated by MARS decision engine 114, I^N neural network decision engine 116, and I^D is generated by dominance-based decision engine 118. Further, each entry $I(i)$, for $i=1, \ldots, N$, can be considered as the (un-normalized) degree to which the case could be classified in rate class i. The last element, $I(N+1)$ indicates the degree to which the case cannot be decided and the entire universe of rate classes is selected.

For illustration purpose, assume that five rate classes are used, *i.e.*, *N*=5, namely:

Rate Class =*{Preferred Best, Preferred, Select, Standardplus, Standard, No Decision (Send to UW)}*

By way of this example, assume that the output of the first classifier (CBE) is: I^C = [0.3, 5.4, 0.3, 0, 0, 0]. This indicates that the second rate class (*e.g.*, Preferred) is strongly supported by the classifier. Normalizing I^C to see the support as a percentage of the overall weights, $\hat{I}^c = [0.05, 0.9, 0.05, 0, 0, 0]$, shows that 90% of the weights is assigned to the second rate class.

Further, to represent partial ignorance, *i.e.*, cases in which the classifier does not have enough information to make a more specific rate classification, discounting may be used. According to an embodiment of the invention, discounting may involve the assignment of some weight to the last element, corresponding to the universe U =(*No Decision: Send to UW*). For example, the previous assignment of I^C could be changed such that I^C = [0.3, 1.4, 0.3, 0, 0, 4], and its normalized assignment would be $\hat{I}^C = [0.05, 0.23, 0.05, 0, 0, 0.67]$. This example shows how 67% of the weights have now been assigned to the universe of discourse U (the entire set of rate classes). This feature allows a representation of the lack of commitment by individual modules. According to an embodiment of the invention, if it is necessary to discount a source because it is not believed to be credible, competent, or reliable enough in generating the correct decision, a portion of the weight is transferred to the universe of discourse (*e.g.*, "any of the above categories"). The determination of the discount may be derived from meta-knowledge, as opposed to object-knowledge. Object knowledge is the level at which each classifier is functioning, *e.g.*, mapping input vectors into decision bins. Meta-knowledge is reasoning about the classifiers' performance over time. Discounting could be static or dynamic. Static discounting may be used a priori to reflect historical (accuracy) performance of each classifier. Dynamic discounting may be determined by evaluating a set of rules, whose Left Hand Side ("LHS") defines a situation, characterized by a conjunct of conditions, and whose Right Hand Side ("RHS") defines the amount by which to discount whichever output is generated by the classifier. According to an embodiment of the invention, postprocessing may be used to detect lack of confidence in a source. When this happens, all the weights may be allocated to the universe of discourse, *i.e.*, refrain from making any decision.

According to an embodiment of the invention, each decision engine model will independently perform a post-processing step. For purposes of illustration, the post processing used for the neural network model will be described. According to an embodiment of the invention, to further improve the classification performance of a neural network module, some post-processing techniques may be applied to the outputs of the individual networks, prior to the fusion process. For example, if the distribution of the outputs did not meet certain pre-defined criteria, no decision needs to be made by the classifier. Rather, the case will be completely discounted by allocating all of the weights to the entire universe of discourse U. The rationale for this particular example is that if a correct decision cannot be made, it would be better not to make any decision rather than making a wrong decision. Considering the outputs as discrete membership grades for all rate classes, the four features that characterize the membership grades may be defined as follows, where N is the number of rate classes and I the membership function, *i.e.*, the output of the classifier.

1. Cardinality

$$\boxed{C=\sum_{1}^{N} I(i)}$$

2. Entropy

$$\boxed{E=\frac{1}{E_{\max}}\sum_{1}^{N} I(i)\times\log(I(i)),\ \text{where } E_{\max}=-\log(1/N)}$$

3. Difference between the highest and the second highest values of outputs.

$$\boxed{D = I_{\max 1} - I_{\max 2}}$$

4. Separation between the rank orders of the highest and the second highest values of outputs

$$\boxed{S = \mathit{RankOrder}(I_{\max 1}) - \mathit{RankOrde}\ r(I_{\max 2})}$$

With the features defined for characterizing the network outputs, the following two-step criteria may be used to identify the cases with weak decisions:

$$\text{Step 1: } C < \tau_1 \ \text{ OR } \ C > \tau_2 \ \text{ OR } \ E > \tau_3$$

$$\text{Step 2: } D < \tau_4 \ \text{ AND } \ S \leq 1$$

where τ_1, τ_2, τ_3, and τ_4 are the thresholds. The value of the thresholds is typically dataset dependent. However, in some embodiments, the value of the thresholds may be independent of the dataset. In the present example related to a neural network classifier module (which in turn is described in greater detail below), the value of the thresholds may be first empirically estimated and then fine-tuned by a global optimizer, such as an evolutionary algorithm. As part of this example, the final numbers are shown below in Table 1. Other optimization methods may also be used to obtain the thresholds.

Thresholds	Non-nicotine Users	Nicotine Users
1	0.50	0.30
2	2.00	1.75
3	0.92	0.84
4	0.10	0.21

Table 1

Thus, post-processing may be used to identify those cases for which the module's output is likely to be unreliable. According to an embodiment of the invention, rather than rejecting such cases, the model assignment of normalized weights to rate classes may be discounted by assigning some or all of those weights to the universe of discourse U.

As described previously, the fusion module 150 may perform the step of determining a combined decision via the associative fusion of the decision engine models' outputs. According to an embodiment of the invention, any general method that can be used to fuse the output of several classifiers may be used. The fusion method may also be associative, meaning that given three or more classifiers, any two of the classifiers may be fused, then fusing the results with the third classifier, and so on, regardless of the order.

By way of example of determining a combined decision, define *m* classifiers S_1, ... S_m, such that the output of classifier S_j is the vector I^j showing the normalized decision of such classifier to the N rate classes. Recall the last $(N+1)^{th}$ element represents the classifier's lack of commitment, *i.e.*, $I^j = [\, I^j(1), I^j(2), \ldots, I^j(N+1)]$, where:

$$I^j(i) \in [0,1] \text{ and } \sum_{i=1}^{N+1} I^j(i) = 1$$

The un-normalized fusion of the outputs of two classifiers S_1 and S_2 is further defined as:

$$F(I^1, I^2) = Outerproduct(I^1, I^2, T) = A$$

where the outer-product is a well-defined mathematical operation, which in this case takes as arguments the two N-dimensional vectors I^1 and I^2 and generates as output the NxN dimensional array A. Each element $A(i,j)$ is the result of applying the operator T to the corresponding vector elements, namely I^1(i) and I^2(j), *e.g.*,

$$A(i,j) = T[\mathbf{I}^1(i), \mathbf{I}^2(j)]$$

and as illustrated in Fig. 2. Matrix 200 illustrates classes 202 and values 204 for vector I^1 and classes 206 and values 208 for vector I^2. Intersection 210 illustrates one intersection between the vector I^1 and vector I^2. Other intersections and representations may also be used.

The operator T(x,y) may be referred to as a Triangular Norm. Triangular Norms (also referred to as "T-norms") are general families of binary functions that satisfy the requirements of the intersection operators. T-norms are functions that map the unit square into the unit interval, *i.e.*, T: [0,1]x[0,1] → [0,1]. T-norms are monotonic, commutative and associative. Their corresponding boundary conditions, *i.e.*, the evaluation of the T-norms at the extremes of the [0,1] interval, satisfy the truth tables of the logical AND operator.

As there appear to be an infinite number of T-norms, the five most representative T-norms for some practical values of information granularity may be selected. According to an embodiment of the invention, the five T-norms selected are:

T-Norm	**Correlation Type**

$$T_1(x,y) = \max(0, x+y-1)$$
$$T_{1.5}(x,y) = \max(0, x^{0.5}+y^{0.5}-1)^2$$
$$T_2(x,y) = x*y$$
$$T_{2.5}(x,y) = (x^{-1}+y^{-1}-1)^{-1}$$
$$T_3(x,y) = \min(x,y)$$

Extreme case of negative correlation

Partial case of negative correlation

No correlation

Partial case of positive correlation

Extreme case of positive correlation

The selection of the best T-norm to be used as an intersection operation in the fusion of the classifiers may depend on the potential correlation among the classifiers to be fused. For example, T3 (the minimum operator) may be used when one classifier subsumes the other one (*e.g.*, extreme case of positive correlation). T2 may be selected when the classifiers are uncorrelated (*e.g.*, similar to the evidential independence in Dempster-Shafer). T1 may be used if the classifiers are mutually exclusive (*e.g.*, extreme case of negative correlation). The operators $T_{1.5}$ and $T_{2.5}$ may be selected when the classifiers show intermediate stages of negative or positive correlation, respectively. Of course, it will be understood by one of ordinary skill in the art that other T-norms may also be used. However, for the purposes of the present invention, these five T-norms provide a good representation of the infinite number of functions that satisfy the T-norm properties.

Because the T-norms are associative, so is the fusion operator, *i.e.*,

$$F(I^1, F(I^2, I^3)) = F(F(I^1, I^2), I^3)$$

Each element *A(i,j)* represents the fused assignment of the two classifiers to the intersection of rate classes r_i and r_j. Fig. 3 illustrates that each rate class is disjointed and that U 300, is the universe of all (rate) classes. In this example, rate classes r_1 302, r_2 304 to r_n 306 are shown. Given that the rate classes are disjoint, there are five possible situations:

(a) When $i=j$ *and* $i< (N+1)$ then $r_i \cap r_j = r_j \cap r_i = r_i$

(b) When $i=j$ *and* $i= (N+1)$ then $r_i \cap r_j = U$ (the universe of rate classes)

(c) When $i \neq j$ *and* $i< (N+1)$ and $j < (N+1)$ then $r_i \cap r_j = \phi$ (the empty set)

(d) When $i \neq j$ *and* $i= (N+1)$ then $U \cap r_j = r_j$

(e) When $i \neq j$ *and* $j=(N+1)$ then $r_i \cap U = r_i$

Fig. 4 depicts a chart 400 that illustrates the result of the intersections of the rate classes and the universe U, according to an embodiment of the invention. The chart demonstrates the intersection according to those situations set forth above, such that when situation (a) occurs, the results are tabulated in the main diagonal identified as 410 in Fig. 4. Further, when situation (b) occurs, the results are tabulated in the appropriate areas identified as 420 in Fig. 4. When situation (c) occurs, the results are tabulated in the appropriate areas identified as 430, while when situations (d) or (e) occur, the results are tabulated in the appropriate areas identified as 440 in Fig. 4. By way of example, when one application is rated r1 in the first instance and r2 in the second instance, the intersection may be tabulated at 450, where the column for r1 and the row for r2 intersect. In this example, the intersection of r1 and r2 is the empty set ϕ. The decisions for each rate class can be gathered by adding up all the weights assigned to them. According to the four possible situations described above, weights may be assigned to a specific rate class only in situation a) and d), as illustrated in Fig. 4. Thus, there will be:

$$Weight\ (r_i) = A(i,i) + A(i,N+1) + A(N+1,i)$$

$$Weight\ (U) = A(N+1,N+1)$$

To illustrate the fusion operator based on T-norms, an example will now be described. Assume that

$I^1 =[0.8, 0.15, 0.05, 0, 0, 0]$ and $I^2 =[0.9, 0.05, 0.05, 0, 0, 0]$

This indicates that both classifiers are showing a strong preference for the first rate class (*e.g.*, "Preferred Best") as they have assigned them 0.8 and 0.9, respectively. Fusing these classifiers using each of the five T-norm operators defined above will generate the corresponding matrices A that are shown in the tables in Figs. 5-9, such that Fig. 5 illustrates an extreme positive correlation, Fig. 6 illustrates a partial positive correlation, Fig. 7 illustrates no correlation, Fig. 8 illustrates a partial negative correlation and Fig. 9 illustrates an extreme negative correlation. If the results are normalized so that the sum of the entries is equal to one, the matrices $\hat{A}$ are generated, as shown in the tables in Figs. 10-14 in a manner corresponding to the un-normalized results. During the process, the un-normalized matrices A (Figs. 5-9) may be used to preserve the associative property. At the end, the normalized matrices $\hat{A}$ are used (Figs. 10-14). Using the expressions for weights of a rate class, the final weights for the N rate classes and the universe U from Figs. 10-14 can be computed. An illustration of the computation of the final weights is illustrated in the chart of Fig. 15. Chart 1500 illustrates the five classes 1510, the five T-norms 1520, and the fused intersection results 1530.

According to an embodiment of the invention, the confidence in the fusion may be calculated by defining a measure of the scattering around the main diagonal. The more the weights are assigned to elements outside the main diagonal, the less is the measure of the consensus among the classifiers. This concept may be represented by defining a penalty matrix $\boldsymbol{P} = [P(i,j)]$, of the form:

$$P(i,j) = \begin{cases} \max(0,(1-W^*|i-j|))^d & \text{for } 1 \le \mathrm{i} \le N \text{ and } 1 \le \mathrm{j} \le N \\ 1 & \text{for } \mathrm{i} = (N+1) \text{ or } \mathrm{j} = (N+1) \end{cases}$$

This function rewards the presence of weights on the main diagonal, indicating agreement between the two classifiers, and penalizes the presence of elements off the main diagonal, indicating conflict. The conflict increases in magnitude as the distance from the main diagonal increases. For example, for W=0.2 and d=5 we have the penalty matrix set forth in Fig. 16. Matrix 1600 intersects the column classes 1610 with the row classes 1620 to determine the appropriate penalty.

Other functions penalizing elements off the main diagonal, such as any suitable non-linear function of the distance from the main diagonal, *i.e.*, the absolute value |i-j|, could also be used. The penalty function is used because the conflict may be gradual, as the (rate) classes have an ordering. Therefore, the penalty function captures the fact that the discrepancy between rate classes r_1 and r_2 is smaller than then the discrepancy between r_1 and r_3. The shape of the penalty matrix *P* in Fig. 16 captures this concept, as *P*1600 shows that the confidence decreases non-linearly with the distance from the main diagonal. A measure of the normalized confidence $\hat{C}$ is the sum of element-wise products between $\hat{A}$ and *P* 1600, *e.g.*:

$$\hat{C} = \text{Normalized Confidence}(\hat{\text{A}},\text{P}) = \sum_{i=1}^{N+1}\sum_{j=1}^{N+1}\hat{A}(i,j) * P(i,j)$$

where $\hat{A}$ is the normalized fusion matrix. The results of the fusion of classifiers S1 and S2, using each of the five T-norms with the associated normalized confidence measure, are shown in Fig. 15.

In a situation in which there is a discrepancy between the two classifiers, this fact may be captured by the confidence measure. For instance, consider a situation different from the assignment illustrated in Figs. 5-14, in which the classifiers agreed to select the first rate class. Now *e.g.*, assume that the two classifiers are showing strong preferences for different rate classes, the first classifier is selecting the second rate class, while the second classifier is favoring the first class:

$$I^1 = [0.15,\ 0.85,\ 0.05,\ 0,\ 0,\ 0] \text{ and } \quad I^2 = [0.9,\ 0.05,\ 0.05,\ 0,\ 0,\ 0]$$

The results of their fusion are summarized in the table of Fig. 17, where the chart 1700 illustrates the rate classes 1710, the T-norms 1720 and the fused intersection results 1730. None of the rate classes have a high weight and the normalized confidence has dropped.

According to an embodiment of the invention, it may be desirable to be able to discount the one of the classifiers, to reflect our lack of confidence in its reliability. For example, the second classifier (S2) in the first example (in which the classifiers seemed to agree on selecting the first rate class) may be discounted:

$I^1 = [0.8, 0.15, 0.05, 0, 0, 0]$ and $I^2 = [0.9, 0.05, 0.05, 0, 0, 0]$

This discounting is represented by allocating some of the classifier's weight, in this instance 0.3, to the universe of discourse U, (U=No decision: Sent_to_UW):

$$I^1 = [0.8, 0.15, 0.05, 0, 0, 0] \quad \text{and} \quad I^2 = [0.6, 0.05, 0.05, 0, 0, 0.3]$$

The results of the fusion of I^1 and I^2 are summarized in Fig. 18 below. Summarization chart 1800 illustrates the classes 1810, T-norms 1820, the fused intersection results 1830 and the confidence measure 1840. The rate classes have a slightly lower weight (for T3, T2.5, T2), but the normalized confidence is higher than with respect to Fig. 15, as there is less conflict. Fusion matrices A are shown in the tables of Figs. 19-23, while the tables of Figs. 24-28 illustrate matrices Â. According to an embodiment of the invention, a fusion rule based on Dempster-Shafer corresponds to the selection of:

a) T-norm operator $T(x,y) = x*y$; and

b) Penalty function using $W=1$ (or alternatively $d = \infty$)

Constraint b) implies the penalty matrix P 2900 illustrated in Fig. 29. Therefore, the two additional constraints a) and b) required by Dempster-Shafer theory (also referred to as "DS") imply that the classifiers to be fused must be uncorrelated (*e.g.*, evidentially independent) and that there is no ordering over the classes, and any kind of disagreement (*e.g.*, weights assigned to elements off the main diagonal) can only contribute to a measure of conflict and not, at least to a partial degree, to a measure of confidence. In DS, the measure of conflict K is the sum of weights assigned to the empty set. This corresponds to the elements with a 0 in the penalty matrix ***P*** 2900 illustrated in Figure 29.

According to an embodiment of the invention, the normalized confidence C described above may be used as a measure of confidence, *i.e.*:

$$\hat{C} = \text{Normalized Confidence}(\hat{\text{A}},\text{P}) = \sum_{i=1}^{N+1}\sum_{j=1}^{N+1}\hat{A}(i,j) * P(i,j)$$

The confidence factor $\hat{C}$ may be interpreted as the weighted cardinality of the normalized assignments around the main diagonal, after all the classifiers have been fused. In the case of DS, the measure of confidence $\hat{C}$ is the complement (to one) of the measure of conflict K, *i.e.*: $\hat{C}$= 1- K, where K is the sum of weights assigned to the empty set.

An additional feature of the present invention is the identification of cases that are candidates for a test set, auditing, or standard reference decision process via the comparison module. As illustrated previously in Fig. 1, the comparison module has four inputs. These inputs include the decision of the production engine, which according to an embodiment of the invention, is one of five possible rate classes or a no-decision (*e.g.*, "send the case to a human underwriter"), *i.e.*:

D(FLE) = r1 and r1 $\in$ {*Best, Preferred, Select, Standardplus, Standard, Sent_to_UW*}

An additional input may comprise the decision of the fusion module, which according to an embodiment of the invention, is also one of five possible rate classes or a no-decision (*e.g.*, "send the case to a human underwriter"), *i.e.*:

D(FUS) = r2 and r2 $\in$ {*Best, Preferred, Select, Standardplus, Standard, Sent_to_UW*}

An additional input may comprise the degree of confidence in the production engine decision. The computation of the confidence measure is described in the U.S. Patent

Application Serial Nos. 10/173,000 and 10/171,575, entitled "A Process/System for Rule-Based Insurance Underwriting Suitable for Use by an Automated System," the contents of which are incorporated herein by reference. This measure may be equated to the degree of intersection of the soft constraints used by a fuzzy logic engine ("FLE"). This measure may indicate if a case had all its constraints fully satisfied (and thus C(FLE) =1) or whether at least one constraint was only partially satisfied (and therefore C(FLE) <1).

An additional input may comprise the degree of confidence in the fusion process. The normalized confidence measure $\hat{C}$ is C(FUS). According to an embodiment of the invention, the first test performed is to compare the two decisions, *i.e.*, D(FLE) and D(FUS). Fig. 30 illustrates all the possible comparisons between the decision of the production engine and the fusion module. Comparison matrix 3000 illustrates the D(FLE) classes 3010 and the D(FUS) classes 3020. From the table it can be seen that label A shows that D(FLE)=D(FUS) and they both indicate the same, specific rate class. Further, label B shows that the fusion module made no automated decision and suggested to send the application to a human underwriter, *i.e.* D(FUS) = No Decision. Label C shows that D(FLE)≠D(FUS) and that both D(FLE) and D(FUS) indicate a specific, distinct rate class. In addition, label D shows that D(FLE)≠D(FUS), and in particular, that the FLE made no automated decision and suggested to send the application to a human underwriter, while the Fusion module selected a specific rate class. Label E shows that D(FLE)=(FUS) and that both D(FLE) and D(FUS) agree not to make any decision.

A second test may be done by using this information in conjunction with the measures of confidence C(FLE) and C(FUS) associated with the two decisions. With this information, the performance of the decision engine may be assessed over time by monitoring the time statistics of these labels, and the frequencies of cases with a low degree of confidence. According to an embodiment of the invention, a stable or increasing number of label A's would be an indicator of good, stable operations. An increase in the number of label B's would be an indicator that the fusion module (with its models) needs to be retrained. These cases might be shown to a team of senior underwriters for a standard reference decision. An increase in the frequency of label C's or of cases with low confidence could be a leading indicator of increased classification risk and might warrant further scrutiny (*e.g.*, auditing, retraining of the fusion models, re-tuning of the production engine). An increase in label D's may demonstrate that either the production engine needs re-tuning and/or the fusion modules needs retraining. An increase in label E's may demonstrate an increase in unusual, more complex cases, possibly requiring the scrutiny of senior underwriters. Thus, the candidates for the auditing process will be the ones exhibiting a low degree of confidence (C(FUS) < T1), regardless of their agreement with the FLE and the ones for which the Fusion and the Production engine disagree, *i.e.*, the ones labeled C.

The candidates for the standard reference decision process are the cases for which the fusion module shows no decisions (labeled B or E). The candidates to augment the test set may be selected among the cases for which the fusion module and the production engine agree (label A). These cases may be filtered to remove the cases in which the production engine was of borderline quality (C(FLE) < T2) and the cases in which the confidence measure of the fusion was below complete certainty (C(FUS) < T1). Thresholds T1 and T2, may be data dependent and must be obtained empirically. By way of example, T1=0.15 and T2=1. Table 2 below summarizes the conditions and the quality assurance actions required, according to an embodiment of the invention. Dashes ("-') in the entries of the table may indicate that the result of the confidence measures are not material to the action taken and/or to the label applied.

Decisions	Confidence Measures		ACTION
Label from Table 7	C(FLE)	C(FUS)	
A	$\geq T2$	$\geq T1$	Candidate to be added to data set for tuning of FLE
B	-	-	Candidate for Stand Ref Dec. Process. After enough cases are collected, re-tune the classifiers
C	-	-	Candidate for Auditing
D	-	-	Candidate for Stand Ref Dec. Process. After enough cases are collected, re-tune the classifiers
E	-	-	Candidate for Stand Ref Dec. Process. After enough cases are collected, re-tune the classifiers
-	-	$< T3$	Candidate for Auditing

Table 2

According to an embodiment of the invention, the fusion module may be implemented using software code on a processor. By way of an example of the results of an implementation of the present invention, a fusion module was tested against a case base containing a total of 2,879 cases. After removing 173 UW cases, the remaining 2,706 cases were segmented into 831 nicotine users, with three rate classes, and 1,875 non-nicotine users, with five rate-classes. These cases were then used to test the fusion process. Because the cases for which the production engine had made no decision were removed, use of a comparison matrix similar to the one of Table 1400 will only have labels A, B, C. The fusion was performed using the T-norm $T2(x,y) = x*y$.

Fig. 31 illustrates the effect of changing the threshold T1 on the measure of confidence $\hat{C}$, were $0 \leq \hat{C} \leq 1$. Table 3100 display decisions 3110, confidence thresholds 3120 and the case distributions 3130 based on the confidence threshold 3120. Each column shows the number of cases whose measure of confidence $\hat{C}$ is ≥ *T1*. As the threshold is raised, the number of "No Fusion Decision" increases. A "No Fusion Decision" occurs when the results of the fusion are deemed too weak to be used. When the threshold T is 1, no case is rejected on the basis of the measure of conflict. This leaves 36 cases for which no decision could be made. As the threshold is decreased, decisions with a high degree of conflict are rejected, and the number of "No Fusion Decisions" increases.

"Agreements" occur when the fused decision agrees with the FLE and with the Standard Reference Decision (SRD). "False Positives" occur when the fused decision disagrees with the FLE, which in turn is correct since the FLE agrees with the Standard Reference Decision ("SRD"). "False Negatives" occur when the fused decision agrees with the FLE, but both the fusion decision and the FLE are wrong, as they disagree with the SRD. "Corrections" occur when the fused decision agrees with the SRD and disagrees with the FLE. Finally, "Complete Disagreement" occurs when the fused decision disagrees with the FLE, and both the fused decision and the FLE disagree with the SRD. Further, similar results were obtained for nicotine users, and these results are illustrated in Fig. 32, with table 3200 displaying decisions 3210, confidence thresholds 3220 and the case distributions 3230 based on the confidence thresholds 3220.

Fig. 33 illustrates a Venn diagram 3300 illustrating the situation for the threshold T1 = 0.15 (*i.e.*, for C≥ 0.15) for the non-nicotine users, while Fig. 34 illustrates a Venn diagram 3400 illustrating the situation for the threshold T1 = 0.15 (*i.e.*, for C≥ 0.15) for the nicotine users. In the case of the non-nicotine users (for *T1* = 0.15) the following labels result:

A: 1,588+27 = 1,615 (86.13%) in which 3310 D(FUS)=D(FLE); (*e.g.*, agreements 3310 and false negative 3320)

B: = 36 (1.92%) in which the fusion did not make any decision (from $\hat{C}$=0);

C1: 212-36 = 176 (9.39%) in which the fusion was too conflictive ($\hat{C}$<0.15); and

C2: 22+25+1= 48 (2.56%) in which D(FUS)≠D(FLE) (*e.g.*, false positive 3330, corrections 3340 and complete disagreements 3350).

In the case of the nicotine users (for *T1* = 0.15), the following labels result:

A: 729+15 = 744 cases (89.5%) in which D(FUS)=D(FLE); (*e.g.*, agreements 3410 and false negatives 3420);

B: = 37 cases (4.5%) in which the fusion did not make any decision (from $\hat{C}$=0);

C1: 68-37 = 31 cases (3.7%) in which the fusion was too conflictive ($\hat{C}$<0.15); and

C2: 16 +3 = 19 cases (2.3%) in which D(FUS)≠D(FLE) (*e.g.*, false positives 3430, corrections 3440 and complete disagreements 3450).

According to the present example, since there is no SRD in production, there can only be reliance on the degree of conflict and the agreement between the fused decision and the FLE. If the disagreement between FLE and FUS (*e.g.*, subset C2) is used, it can be observed that the number of cases in which the fusion will disagree with the FLE, and make a classification, is 48/1875 (2.56%) for non-nicotine users and 19/831 (2.3%) for nicotine users. This may be considered a manageable percentage of cases to audit. Further, this sample of cases may be augmented by additional cases sampled from subsets C1.

A further analysis of set C2 in the case of non-nicotine users shows that out of 48 cases, the fusion module called 22 of them correctly and 26 of them incorrectly. From the 26 incorrectly called cases, 14 cases were borderline cases according to the FLE. This illustrates that the problematic cases may be correctly identified and are good candidates for an audit.

A further analysis of set C2 in the case of nicotine users shows that out of 19 cases, the fusion module incorrectly called 16. Of these 16 cases, 6 cases were borderline cases, *i.e.*, the FLE only had partial degree of satisfaction of the intersection of all the constraints *e.g.*, C(FLE)<0.9. Furthermore, 11 cases had a conflict measure $\hat{C} < 0.4$. If the union of these two subsets (*e.g.*, the borderline cases and the conflict measure cases) is taken, the results are 13 cases that are either borderline (from the FLE) or have low confidence in the fusion, and the remaining 3 cases were ones that the CBE could not classify (*i.e.*, it could not find enough similar cases). This again demonstrates that the problematic cases may be generally correctly identified and are worth auditing.

The set B (4.5%) illustrates a lack of commitment and is a candidate for a review to assign an SRD. The set A may be a starting point to identify the cases that could go to the test set. However, set A may need further filtering by removing all cases that were borderline according to the FLE (*i.e.*, C(FLE)< T2), as well as removing those cases whose fusion confidence was too low (*i.e.*, C(FUS) <1). Again T2 will be determined empirically, from the data.

Various aspects of the fusion module will now be discussed in greater detail below. It is understood that various portions of the fusion module, as well the different aspects described below, may be performed in different manners without departing from the scope of the invention.

2. Outlier Detector

One component of a fusion module may be determining outlier applications. According to an embodiment of the invention, it may be desirable to detect all classification assignments to applications, such as insurance applications, that are inconsistent and therefore potentially incorrect. Applications that are assigned these inconsistent labels may be defined as outliers. The concept of outliers may extend beyond the realm of insurance underwriting and be intrinsic to all risk classification processes, of which the determination of the proper premium to cover a given risk (*i.e.*, insurance underwriting) is just an example. Therefore, the ultimate domain of this invention may be considered risk classification, with a focus on insurance underwriting.

According to an embodiment of the invention, the existing risk structure of the risk classification problem is exploited from the risk assignments made by the underwriters, similar to the dominance-based classifier described in greater detail below. But whereas the dominance based classifier uses the risk structure to produce a risk assignment for an unlabeled application, the outlier detector examines the risk structure to find any applications that might have been potentially assigned an incorrect risk assignment by the underwriter.

The outlier detector may add to the rationality of the overall underwriting process by detecting globally inconsistent labels and bringing it to the attention of human experts. Many papers in the decision sciences demonstrate that in the presence of information overload, humans tend to be boundedly rational and often, unintentionally, violate compelling principles of rationality like dominance and transitivity. The outlier detector may attempt to counter these drawbacks exhibited by human decision-makers and make the decision-making process more rational. As a result, the risk assignments can be expected to be more optimal and consistent.

Further, by bringing these globally inconsistent risk assignments to the attention of the underwriters, the system may gain knowledge about exceptional decision rules, or additional features that are implicitly used by experts and which may be left unmentioned during the initial design stages of an automated system. This additional knowledge may be used to improve the performance of any automated system. Thus, the outlier detector may also act as a knowledge-eliciting module.

By removing globally inconsistent risk assignments from the initial set, the detection of outliers may further improve the performance and simplicity of other supervised classification systems, such as neural networks and decision-tree classifiers when used as the primary automated system. This is because the presence of global inconsistencies may add to the "non separability" of the feature space, which will often lead to either inferior learning, or very complicated architectures. As the outlier detector reduces the number of global inconsistencies, a cleaner, more consistent training set may be expected to result in a better learning, and by a simpler system. Hence, the outlier detector may improve the classification accuracy, and simplicity of other automated systems.

Because the outlier detector uses the principle of dominance to capture the risk structure of the problem, the outlier detector has explanation capability to account for its results. This is because dominance is a compelling principle of rationality and thus the outliers detected by the system are rationally defensible.

According to an embodiment of the invention, the functionality of the outlier detection system may be generic, so that it can be used to detect outliers for any preference-based problem where the candidates in question are assigned preferences based on the values that they take along a common set of features, and the preference of a candidate is a monotonic function of its feature-values. Therefore, the applicability of an outlier detection system transcends the problem of insurance underwriting, and can be easily extended to any risk classification process.

In many domains where expert opinions are used to score entities, the set of entities that have already been scored are stored as precedents, cases, or reference data points for use in future scoring or comparison with new candidates. The outlier detector can help in ensuring that any new candidate case that goes into the reference dataset will always lead to a globally consistent dataset, thereby ensuring that the reference dataset is more reliable.

According to an embodiment of the invention, an outlier detector may exploit the existing risk structure of a decision problem to discover risk assignments that are globally inconsistent. The technique may work on a set of candidates for which risk categories have already been assigned (*e.g.*, in the case of insurance underwriting, for example, this would pertain to the premium class assigned to an application). For this set of labeled candidates, the system may find all such pairs of applications belonging to different risk categories, which violate the principle of dominance. The outlier detector attempts to match the risk ordering of the applications with the ordering imposed by dominance, and use any mismatch during this process to identify applications that were potentially assigned incorrect risk categories.

As described previously, automating an insurance underwriting process may involve trying to emulate the reasoning used by the human expert while assigning premium classes to insurance applications, and finding computable functions that capture those reasoning principles. According to an embodiment of the invention, the risk category of an application depends upon the values taken by the application along various dimensions, such as Body Mass Index ("BMI"), Cholesterol Level, and Smoking History. The values of the dimensions are then used to assign risk categories to insurance applications. An automated system would operate on these same features while trying to emulate the underwriter. Typically, the risk associated with an application changes with changes to the magnitude of the individual features. For example, assuming that all other features remaining the same, if the BMI of an applicant increases, the application becomes riskier. The outlier detector uses this knowledge to detect all such applications that do not satisfy the principle of dominance.

According to an embodiment of the invention, there is a monotonic non-decreasing relationship between all the feature-values and the associated risk (*e.g.*, higher values imply equal-or-higher risk). Variables that do not meet this relationship may be substituted by their mirror image, which will then satisfy this condition. For instance, let us assume that the relevant medical information for a non-smoker applicant is captured by the following five variables:

X1 = *Cholesterol,*

X2 = Cholesterol Level,

X3 = Systolic Blood Pressure,

X4 = Diastolic Blood Pressure,

X5 = Years since quitting smoking (if applicable).

Mortality risk is monotonically non-deceasing with respect to the first four variables, meaning that such risk can increase (or remain the same) as the values of the four variables increase. However, higher values in the fifth variable have a positive effect, as they decrease the mortality risk. Therefore, the fifth variable needs to be transformed into another variable. By way of example, X5 may be transformed into X5', where X5' is defined as X5'=K - X5 = K-"years since quitting smoking". K is a constant, *e.g.*, K=7, so that higher values of X' will reflect same or increased mortality risk. Other relationships between all the feature-values may also be used.

Further, if two insurance applicants A and B are compared where applicants A and B are identical along all features, except that the applicant B has a higher BMI than A, then the risk associated with applicant A cannot be greater than that associated with B. In other words, the premium associated to the rate class assigned to A should not be higher than that one assigned to B. The above reasoning principle is referred to, in decision theory, as the principle of dominance and in the above example applicant A dominates applicant B. The terminology *dominates(A,B)* is used to capture this relation between applicant A and applicant B.

For example, given two applications A and B, it can be said that application A dominates application B if and only if application A is at least as good as application B along all the features and there is at least one feature along which application A is strictly better than application B. The *dominates* relation may be based on the above definition of dominance. It is a trichotomous relation, meaning that given two applications A and B either application A dominates application B, application B dominates application A, or neither dominates the other. In the case where neither applicant dominates the other, each application may be better than its counterpart along different features. In such a case, application A and application B may be said to be dominance-tied. For example, as illustrated in Table 3 below, assume there are three applicants A, B, and C with the following feature values:

Application	**BMI**	**Cholesterol**	**BP_sys**
A	25	255	115
B	26	248	120
C	24	248	112

Table 3

Assuming for simplicity that these are the only three features used to assess the risk of an applicant. By the definition, it can be seen that application C dominates both application A and application B, since application C is at least as good (*e.g.*, as low) as application A and application B along each feature, and moreover there is at least one feature along which application C is strictly better (*e.g.*, strictly lower) than both application A and application B. However, application A and application B are dominance-tied since each is better (*e.g.*, lower) than the other along some feature (application A has better cholesterol value while application B has better BMI value).

According to an embodiment of the invention, the relation *No_Riskier_Than(A,B)* is true if the risk associated with applicant A (say r_A) is no higher than that associated with applicant B (say r_B), *i.e.*,

$$\textit{No\_Riskier\_Than(A,B)} \leftrightarrow (r_A \leq r_B).$$

According to an embodiment of the invention, based on the assumption that the risk associated with an applicant is a monotonic non-decreasing function of the feature values, it can be seen that for any pair of insurance applications, if the dominates relation holds between the two applications in a certain direction (*e.g.*, application A dominates application B), then the No_Riskier_Than relation will also hold in the same direction (*e.g.*, application A is *No_Riskier_Than* application B). In other words, the *dominates* relation is a sufficiency condition for the *No_Riskier_ Than* relation. That is:

$$\textit{dominates(A,B)} \rightarrow \textit{No\_Riskier\_Than(A,B)}.$$

An application may be considered an outlier based on one or more characteristics. According to an embodiment of the invention, application X and application Y are marked as outliers if application X dominates application Y, and application X is assigned a risk category that associates greater risk with application X compared to application Y. According to an embodiment of the invention, application X and application Y are marked as outliers if application Y dominates application X, and application Y is assigned a risk category that associates greater risk with application Y compared to application X.

The above statements can be described formally with the following equation:

$$(\text{X,Y are outliers}) \leftrightarrow (\,\textit{dominates(X,Y)} \wedge (\,r_X > r_Y)\,)$$

$$\vee(\,\textit{dominates(Y,X)} \wedge \;(r_Y > r_X)\,)$$

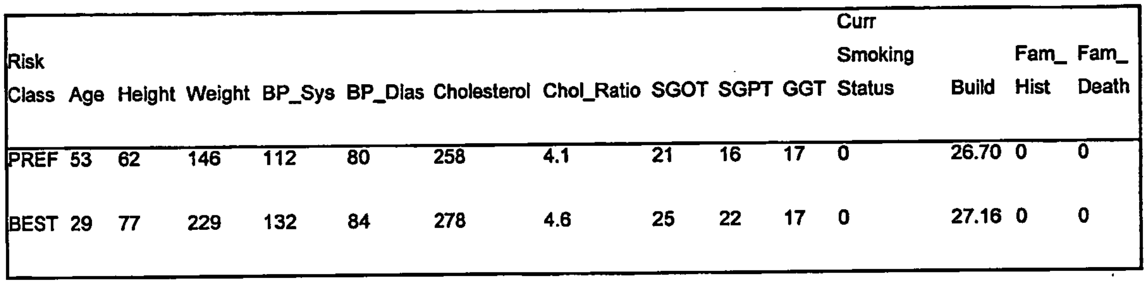
As can be seen, from the definitions of the *dominates* relation and the *No_Riskier_Than* relation, inconsistent risk assignments may be identified. If application X dominates application Y, then application X will be at least as good as application Y along all features and strictly better than application Y along at least one feature. As a result, logically, application X cannot be riskier than application Y. Therefore, if the risk assignments made by the underwriters are such that application X is categorized as being riskier than application Y, then the existing risk assignments made to application X, and application Y, or to both application X and application Y, may likely be logically infeasible. Therefore, both application X and application Y are labeled as outliers, *e.g.*, applications that have inconsistent assignments, and therefore potentially incorrect risk categories. According to an embodiment of the invention, in order to exploit the presence of the dominance relation between two applications and to logically restrict the risk assignment of the two applications, it may be necessary to ensure that all the features that are being used by the experts during the risk assignments are also used during the dominance comparisons.

The steps involved in outlier detection according to an embodiment of the invention are described below and shown in Fig. 35. An outlier module operates on a set A of applications, each of which has been assigned a risk category from one of the *i* possible categories. The system may be thought of as operating on a set of tuples $\{(A_j,x)\}$ where *x* is the risk category assigned by the underwriter to application A_j. The process for outlier detection may be implemented in pseudocode as set forth below:

```
Outlier_detect(A:{$A_j$,x})
{
for each tuple ($A_j$,x) $\in$ A
{
    for each tuple ($A_k$,y) $\in$ A where $r_y > r_x$
    {
      if (dominates($A_k$,$A_j$))
          mark $A_j$, $A_k$ as outliers;
          break;
      else
          next $A_k$;
    }
next $A_j$;
}
Report set of outliers;
}
```

As defined earlier, outliers are pairs of tuples (A_p,x), (A_q,y) where A_p dominates A_q but $r_y < r_x$. Fig. 35 illustrates a flowchart for detecting outliers given a set of labeled applications. At step 3510, a tuple (A_i,x) is identified. A tuple (A_j,y) is identified at step 3520, where the rate class r_y for tuple (A_j,y) is greater than the rate class r_x. At step 3530, a determination is made whether tuple (A_j,y) dominates tuple (A_i,x) (*e.g.*, *Dominates* $((A_j,i))$. If yes, tuples (A_j,x) and (A_j,y) are marked as outliers. The system then determines at step 3550 if there is another tuple (A_j,y), where $r_y > r_x$. This determination is also made if tuple (A_j,y) does not dominate tuple (A_i,x). At step 3550, if there is another (A_j,y), where $r_y > r_x$, the process returns to step 3520. If there is no other tuple (A_j,y) where $r_y > r_x$, a determination is made at step 3560 whether there is another tuple (A_i,x). If yes, the process returns to step 3510, while if not, the system ends at 3570.

According to an embodiment of the invention, an outlier detector may be implemented in software code, and tested against a database of cases. For example, an outlier detector may be tested against a database of approximately 2,900 cases. In such an example, the outlier detector identified more than a dozen of subsets containing at least one inconsistency. The results produced by the outlier detector in this example are shown in Table 4 below, along with a few relevant feature values.



Risk Class	Age	Height	Weight	BP_Sys	BP_Dias	Cholesterol	Chol_Ratio	SGOT	SGPT	GGT	Curr Smoking Status	Build	Fam_ Hist	Fam_ Death
PREF	53	62	146	112	80	258	4.1	21	16	17	0	26.70	0	0
BEST	29	77	229	132	84	278	4.6	25	22	17	0	27.16	0	0

Table 4

In Table 4 above, each row represents an insurance application for which the risk classification had already been determined, as shown in the first column. The risk class "BEST" is a lower risk class compared to the risk class "PREF." A person classified in the "BEST" risk class will have to pay a lower premium than a person classified in the "PREF" class. Yet, it can be seen that the application indicated in the row first of Table 4 dominates the application of the second row. In the present example, upon sending these two applications to human underwriters for reconsideration, the risk classifications for the applications were reversed. This simple example illustrates the use of an outlier detector to obtain more consistent risk assignments.

As illustrated in Fig. 1 above, outlier detector 180 is shown after the fusion to insure that any new addition to the best-cases database would be dominance-consistent with the existing cases. Another potential use for the outlier detector is its application to the training-cases database used to train each of the decision engines used by the fusion module. This is a Quality Assurance step for the training data to insure that the training cases do not contain outliers (*e.g.*, inconsistent cases in the dominance sense) so as to improve the learning phase of the four models illustrated (CBR, NN, MARS, Dominance) before they are used as run-time classifiers for the Quality Assurance process of the production engine. According to an embodiment of the invention, as illustrated in Fig. 36, an outlier detector 3610 and a training case-base 3620 may be positioned for quality assurance for CBR DE 3630, MARS DE 3640, NN DE 3650 and DOM DE 3660, the output of which is fed into a fusion module (not shown).

3. Dominance Classifier

According to an embodiment of the invention, the risk structure of an underlying problem may also be exploited to produce a risk category label for a given application, such as an insurance application. This risk classification can be assured to be accurate with a high degree of confidence. Specifically, as described above in relation to the outlier detector, the application of a dominance classifier may also provide risk assignments having a high confidence measure. Further, when strict definitions are implemented, the relative accuracy of the system approaches 100%, thus minimizing the degree of mismatch between the risk assignment made by a human underwriter and the automated rate class decisions.

A dominance classifier may have many of the advantages of the outlier detector. The principle of dominance is a compelling principle of rationality and thus the classification produced by the technique is rationally defensible. This imparts explanation capability to the classification making it transparent and easy to comprehend. Further, there are no iterative runs involved in tuning. As a result, the tuning process may reduce and become less time-consuming. The output of this dominance-based classifier can be combined in a fusion module with the output(s) generated by other classifiers. A fusion process may be used for quality assurance of a production decision engine, to provide a stronger degree of confidence in the decision of the engine, in the case of consensus among the classifiers, or to suggest manual audit of the application, in the case of dissent among the classifiers.

According to an embodiment of the invention, automating an insurance application underwriting process may essentially involve trying to emulate the reasoning used by a human expert while assigning premium classes to insurance applications, and finding computable functions that capture those reasoning principles. The risk category of an application depends upon the values taken by the application along various dimensions, such as, but not limited to, body mass index (BMI), cholesterol level, and smoking history. An underwriter makes use of these values to assign risk categories to the applications. Hence, an automated system should operate on these same features while trying to emulate the underwriter. Typically, the manner in which the risk associated with an insurance application changes with changes to the magnitude of the individual features is also known. For example, when all other features in an insurance application remain the same, if the BMI of an applicant increases, the application becomes riskier.

A dominance-based risk classification may use this knowledge to generate a risk category for a given application, such as an insurance application. According to an embodiment of the invention, an assumption may be made that there is a monotonic non-decreasing relationship between all the feature-values and the associated risk (*i.e.*, higher values imply equal-or-higher risk). For those variables that do not meet this relationship, a mirror image may be substituted, which will then satisfy this condition that lower values correspond to lower risk. This can be seen with reference to Table 3 regarding the outlier detector.

Further, as discussed above with respect to the outlier detector, the relation: *dominates (A,B) →No_Riskier Than (A,B)* still holds

The term *Bounded_within(B,{A,C})* may be used when application B is bounded_within application A and application C, if and only if application A dominates application B and application B dominates application C, *i.e.*,

$$\textit{Bounded\_within}(B,\{A,C\}) \leftrightarrow \textit{dominates}(A, B) \wedge \textit{dominates}(B,C).$$

This relation may then be read as "B is bounded within A and C."

If application B is bounded within two applications A and C, and if the risk category assigned to applications A and C is the same, then the risk category of application B has to be the same as that of applications A and C. *i.e.,*

$$\textit{Bounded\_within}(B,\{A,C\}) \wedge (r_A = r_C = r) \rightarrow (r_B = r)$$

To better demonstrate this, suppose the following is present:

$$\textit{Bounded\_within}(B,\{A,C\}) \wedge (r_A = r_C = r).$$

This implies that

$$\textit{dominates}(A, B) \wedge \textit{dominates}(B,C) \wedge (r_A = r_C = r).$$

Or,

$$\textit{No\_Riskier\_Than}(\mathrm{A,B}) \wedge \textit{No\_Riskier\_Than}(\mathrm{B,C}) \wedge (r_A = r_C = r).$$

Based on the definitions of the relation, the above can be rewritten as,

$$(r_A \leq r_B) \wedge (r_B \leq r_C) \wedge (r_A = r_C = r).$$

In other words,

$$r_B=r$$

thereby demonstrating the principle of dominance based risk classification.

This principle may serve as the basis for a risk classification. For any given application B with unassigned risk category, a determination is made whether there exist two applications A and C such that the Left Hand Side (LHS) of the principle is satisfied, *i.e.*, *Bounded_within(B,{A,C})* $\wedge$ ($r_A = r_C = r$). If this occurs, the risk category of application B is assigned to be the same as that of applications A and C.

Even if an application A dominates another application B, the two applications may still be quite close in terms of their feature-values so that they belong to the same risk category. In other words, it may be expected for the *dominates* relation to hold between some pairs of applications even if the two applications belong to the same risk category. This may mean that further partitions of the applications within a risk category may be made, such as into the best, non-dominated subset and worst, non-dominating subset.

According to an embodiment of the invention, the best, non-dominated subset for a given risk category may be defined as the one that contains all such applications that are not dominated by another application within that risk category. This may also be referred to as the Pareto-best subset.

According to an embodiment of the invention, the worst, non-dominating subset for a given risk category may be defined as the one that contains all those applications that do not dominate even a single application in that risk category. This may also be referred to as the Pareto-worst subset.

To visualize these two subsets geometrically, Fig. 37 may be referred to, which shows a plot of features f1 3710 and f2 3720 for 1000 insurance applications. The insurance applications are plotted as points in the 2-dimensional feature space. For simplicity, assume that these are the only two features used while assigning a risk category to the applications, and that the lower values along a feature correspond to a lower risk. In Fig. 37, circles denote the Pareto-best subset 3730 while the squares denote the Pareto-worst subset 3740. The circles take the lowest (*e.g.*, the most desirable) values along both features while the squares take on the highest (*e.g.*, the least desirable) values. In addition, using the definition of the Pareto-best subsets 3730 and the Pareto-worst subsets 3740 as set forth above, each of the remaining insurance applications is such that at least one application represented by a circle dominates it, and it dominates at least one application represented by a square. In other words, for each point *X* that is not in the Pareto-best subset(O) 3730 or in the Pareto-worst subset(P) 3740 in Fig. 37, there is at least one square *S* and one circle *C* such that Bounded_within(X,{C,S}) is true. For example, suppose that every circle and square in Fig. 37 representing an application was assigned the same risk category *r*. Then, by applying the principle of dominance-based risk classification, all the points shown in Fig. 37 can be assigned the risk category *r* as well.

According to an embodiment of the invention, the production of the two subsets *O* and *P* is identical to the production of the dominance subset in discrete alternative decision problems. By way of example, articles by Kung, Luccio, and Preparata (1975), and Calpine and Golding (1976), the contents of which are incorporated herein by reference, present algorithms which can create these subsets in $O(n \, . \, log^{m-1}(n))$ time, where *n* is the number of candidates involved and *m* is the number of features along which the dominance comparisons are being done. Hence, for an underwriting problem with *r* risk categories, there may be *2r* such subsets, or one pair for each risk category representing the risk surfaces that form the upper bound and the lower bound.

According to an embodiment of the invention, an algorithm may produce the Dominance subset for a given set of alternatives X(n,m) where *n* is the number of candidates and *m* is the number of features used. The term Dominance(X,k) may be used to indicate the application of such an algorithm to the set X(n,m), where k is either +1 or –1, depending upon whether higher or lower feature values are desired to be considered as better during dominance comparisons. According to an embodiment of the invention, two principal modules, the tuning module and the classification module, may be used. The tuning module may compute the Pareto-best and Pareto-worst subsets for each risk category. The Classification module may use the results of the tuning to classify new applications.

The tuning module may use the Dominance algorithm to compute the Pareto-best and the Pareto-worst sets for each risk category. Given a set of applications *A*, such as insurance applications that have been partitioned into *i* different risk categories by the underwriter, tuning may use the pseudocode set forth below:

```
TUNE(A,i){
for each risk category $r_i$
            {
            Compute and store the indices of the Pareto-Best subset $O(r_i)$.
                Obtain the Dominance(A) enforcing that lower feature-values are better.
            Compute and store the indices of the Pareto-Worst subset $P(r_i)$.
                Obtain the Dominance(A) enforcing that higher feature-values are better.
            }
```

Fig. 38 is a flowchart illustrating the steps involved in the tuning process according to an embodiment of the invention. At step 3800, each separate risk category is determined. At step 3802, a set of applications A is divided into the different risk categories. At step 3804, the Pareto-best subset of the applications within each risk category is computed. At step 3806, the Pareto-best subset is stored. At step 3808, the Pareto-worst subset of the applications within each risk category is computed. At step 3810, the Pareto-worst subset is stored, completing the tuning process at step 3812.

The classification module may use the sets *O* and *P* from the tuning process to assign risk classifications to new applications. According to an embodiment of the invention, the classification module assigns a risk category to any new application by checking if a given application satisfies the *Bounded_within* relation with respect to a Pareto-best, and another Pareto-worst application for a given rate class. According to an embodiment of the invention, given a set of unlabeled applications, U, and the Pareto-best subsets and the Pareto-worst subsets obtained for each of the *i* risk categories from tuning, each application in U is assigned a risk category. Assignment of a risk category may be carried out according to the pseudocode set forth below using the principle of dominance based risk classification:

Fig. 39 illustrates the steps involved in the classification process according to an embodiment of the invention. At step 3902, an application is selected from U. At step 3904, a risk category r_k is selected. At step 3906, a determination is made whether application Z is bounded within some $x \in O(r_k)$, $y \in P(r_k)$. If not, a determination is made if there is another risk category r_k, at step 3908. If there is another r_k, the process returns to step 3904. If there is no other r_k, application Z is declared unresolved at step 3910, and a determination is made if there is another application Z at step 3912. If there is another application Z, the process returns to step 3902. If there is no other application Z, the process ends at step 3916.

Returning to step 3906, if application Z is bounded, risk category r_k is assigned to application Z at step 3914. The process then moves on to step 3912 to determine if there is another application Z.

When assigning a risk category, such as according to the pseudocode steps illustrated previously or according to the steps of Fig. 39, there may be situations that need to be accounted for in the above risk assignment algorithm. One example is where there is no risk category for which the *Bounded_within* condition is satisfied for A[j]. Another example is where there are at least two risk categories for which the *Bounded_within* condition is satisfied for A[j]. Each of the above two situations can lead to a different kind of ambiguity. Other situations may also lead to various types of ambiguity.

According to an embodiment of the invention, where there is no risk category for which the *Bounded_within* condition is satisfied for A[j], an application may be regarded as ambiguous by the system. No risk category is assigned to the application and the application is marked as unresolved.

The comparison matrix 4000 illustrated in Fig. 40 provides an example of the performance of the system for a particular set of applicants. In the example illustrated in Fig. 40, the system initially used the tuning set in order to compute the Pareto-best and the Pareto-worst subsets for each of the risk categories, which in this case are eight risk categories. The system may then classify a set of applications that were not in the tuning set. For these applications, risk assignments were also obtained from the human underwriters. This allows a comparison of the performance of the system with that of the experts using the comparison matrix.

As mentioned earlier, an application that does not satisfy the *Bounded_within* relation for any of the risk categories, is marked as unresolved by the system. These applications are shown in the column 4002 labeled "UW." As can be seen, quite a large number of applicants were marked as unresolved by the system. However, for the applications that were assigned a risk category by the system, the system was accurate 100% of the time. Thus, 52 applications were correctly classified in column 4004 labeled "PB," 22 applications were correctly classified in column 4006 labeled "P," 16 applications were correctly classified in column 4008 labeled "Sel," 10 applications were correctly classified in column 4010 labeled "Std+," 3 applications were correctly classified in column 4012 labeled "Std," 28 applications were correctly classified in column 4014 labeled "P Nic," 8 applications were correctly classified in column 4016 labeled "Std+Nic," and 3 applications were correctly classified in column 4018 labeled "Std Nic." Hence, the principle of dominance based risk classification presented in this letter has the potential to produce risk assignments with a high degree of confidence. For the few applications that are misclassified above, the use of another system called the dominance based outlier detection system may be used. The dominance based outlier detection system has been described above.

As can be seen from the example of Fig. 40, the classifier is 100% accurate, but may have a lower coverage, meaning that it does not provide a decision for a large number of cases. A different tradeoff may be achieved between relative accuracy and coverage of the system by allowing a minor relaxation of the classification rule used in the extreme rate classes (*e.g.*, the best and worst rate class). According to an embodiment of the invention, one type of modification makes use of the fact that since the risk categories are totally ordered, the principle of dominance-based risk classification can be relaxed for the best and the worst risk categories. This relaxation may therefore be expected to improve the coverage of the automated system. The basis for this relaxation principle may be seen from understanding that if the application for applicant X dominates the application for applicant A such that the risk category assigned to application A is the best risk category for the problem, say r_{best}, then the risk category of application X is also r_{best}, *i.e.*:

$$\mathit{dominates(X,A)} \wedge (\mathrm{r}_{\mathrm{A}} = r_{best}) \rightarrow (\mathrm{r}_{\mathrm{X}} = r_{best}).$$

For example, assume that there is an application X such that it dominates application A, where it is known that A is assigned the best risk category, *i.e.*:

$$rA=rbest$$

Since application A belongs to the best risk category, no other applicant can be assigned a better risk category than application A. In other words,

$$rX \geq rA$$

However since application X also dominates application A, application X can be no riskier than application A which implies that:

$$rX \leq rA$$

From this, it can therefore be inferred that:

$$rX = rbest$$

thereby demonstrating the applicability of the relaxation condition described above with respect to the best classification. Further, the relaxed principle of dominance based risk classification for the worst risk category can be seen by noting that if application A dominates application X such that the risk category assigned to application A is the worst risk category, say r_{worst}, then the risk category of application X is also r_{worst}; *i.e.*:

$$dominates(A,X) \wedge (r_A = r_{worst}) \rightarrow (r_X = r_{worst}).$$

For example, assume that there is an application X such that it is dominated by application A, where it is known that A is assigned the worst risk category. *i.e.*:

$$r_A = r_{worst}$$

Because application A belongs to the worst risk category, every other applicant belongs to a risk category that is better than or equal to that of application A. In other words:

$$r_X \leq r_A$$

However, since application A also dominates application X, therefore application A must also be no riskier than application X, which implies that:

$$r_X \geq r_A$$

From this, it is demonstrated that:

$$r_X = r_{worst}$$

thereby demonstrating the applicability of the relaxation condition described above with respect to the worst classification. Thus, according to an embodiment of the invention, the steps for classification remain the same except that during the r_k-loop in Fig. 39, the application at hand is tested for the relaxed conditions described above respectively, and assigned the risk category accordingly if one of the conditions is satisfied.

The comparison matrix 4100 shown in Fig. 41 illustrates performance of the dominance based risk classifier used after incorporating the relaxed conditions defined above, during classification of an applicant and tested against a case base of approximately 541 cases. Coverage of the classifier has improved, since 68 applicants that were initially marked as unresolved by the classifier are now assigned a risk category. Whereas the relative accuracy of the new classifier is not 100% like its counterpart, the number of misclassifications is relatively few. In other words, for a large gain in coverage the overall drop in accuracy obtained by the use of the modified classifier may be relatively minor. Thus, the relaxation conditions may permit a tradeoff between accuracy and coverage of the dominance based risk classifier. Where the relative accuracy is more important for a problem, the earlier version of the classifier may be used. On the other hand, if some problem requires that more applicants be assigned a risk category, then it may be more desirable to use the modified classifier. This imparts flexibility to the system on the whole since it can cater to varying requirements of accuracy and coverage from the automated system, which is an added advantage of the system.

4. Multivariate Adaptive Regression Splines

According to an embodiment of the invention, a network of multivariate adaptive regression splines ("MARS") based regression models may be used to automate decisions in business, commercial, or manufacturing process. Specifically, such a method and system may be used to automate the process of underwriting an application as applicable to the insurance business.

According to an embodiment of the invention, a MARS based system may be used as an alternative to a rules-based engine ("RBE"). U.S. Patent Application Serial Nos. 10/173,000, filed on June 18, 2002, and 10/171,575, filed on June 17, 2002, titled "A Method/System of Insurance Underwriting Suitable for Use By An Automated System," the contents of which are incorporated herein by reference in their entirety, describe a fuzzy rule-based system. A MARS model may not be as transparent as other decision engines (*e.g.*, "RBE"), but may achieve better accuracy. Therefore, MARS may be used as an alternative approach for a quality assurance tool to monitor the accuracy of the production decision engine, and flag possible borderline cases for auditing and quality assurance analysis. Further, a MARS module may be a regression-based decision system, which may provide the simplicity of implementation of the model since it is based on a mathematical equation that can be efficiently computed.

According to an embodiment of the invention, a MARS module may facilitate the automation of the "clean case" (*e.g.*, those cases with no medical complications) underwriting decision process for insurance products. A MARS module may be used for other applications as well. A MARS module may be used to achieve a high degree of accuracy to minimize mismatches in rate class assignment between that of an expert human underwriter and the automated system. Further, the development of a parallel network of MARS models may use a set of MARS models as a classifier in a multi-class problem.

The MARS module is described in the context of a method and system for automating the decision-making process used in underwriting of insurance applications. However, it is understood that the method and system may be broadly applicable to diverse decision-making applications in business, commercial, and manufacturing processes. Specifically, a structured methodology based on a multi-model parallel network of MARS models may be used to identify the relevant set of variables and their parameters, and build a framework capable of providing automated decisions. The parameters of the MARS-based decision system are estimated from a database consisting of a set of applications with reference decisions against each application. Cross-validation and development/hold-out may be used in combination with re-sampling techniques to build a robust set of models that minimize the error between the automated system's decision and the expert human underwriter. Furthermore, this model building methodology may be used periodically to update and maintain the family of models, if required, to assure that the family of models is current.

Fig. 42 is a flowchart illustrating a process for building a MARS module according to an embodiment of the invention. At step 4205, one or more applications (also referred to as cases) are digitized. Digitization may include assuring that the key application fields required by the model to make a decision are captured in digital form by data entry.

In step 4210, a case base is formed. Creating a case base may include assuring that the records corresponding to each application (*e.g.*, case) are stored in a Case Base (CB) to be used for model construction, testing, and validation. In step 4215, preprocessing of cases occurs. Preprocessing may include one or more sub-steps. By way of example, preprocessing may involve location translation and truncation 4216, such as focusing on values of interest for each field. Further, preprocessing may involve range normalization 4217, such as normalizing values to allow for comparison along several fields. Preprocessing may also involve tag encoding 4218, where tag encoding includes augmenting a record with an indicator, which embodies domain- knowledge in the record by evaluating coarse constraints into the record itself.

In step 4220, partitioning and re-sampling occurs. According to an embodiment of the invention, five-fold partitioning may be used, with a stratified sampling within each rate class used to create five disjoint partitions in the CB. In step 4225, generation of a development and validation set occurs. Each partition may be used once as a validation set, with the remaining four used as training sets. This may occur five times to achieve reliable statistics on the model performance and robustness.

At step 4230, one or more model building experiments occur. Experiments with modeling may involve modeling techniques such as global regression and classification and regression trees ("CART") to determine rate classes from a case description. This may result with the selection of MARS as the modeling paradigm.

At step 4235, a parallel network of MARS models is implemented. According to an embodiment of the invention, implementation of networks of MARS models may be used to improve classification accuracy.

According to an embodiment of the invention, the MARS model(s) described may be used as an input to a fusion module. Fusion of multiple classifiers based on MARS, Case-based Reasoning, Neural Networks, *etc.*, may be used to improve classification reliability, as described above. The steps of the process illustrated in Fig. 42 will now be described in greater detail.

At step 4205, cases are digitized and at step 4210, a case base is formed. According to an embodiment of the invention, a MARS model framework starts from a database of applications with the corresponding response variable (*e.g.*, rate class decisions) provided for each. This may be done via cooperative case evaluation sessions with experienced underwriters, or may be accomplished via the reuse of previously certified cases. This database of applications is hereby referred to as a "Certified Case Base" or a "Case Base". According to an embodiment of the invention, it is assumed that the characteristics of the certified case base closely match those of incoming insurance applications received in a reasonable time window *i.e.*, they form a "representative sample." The Case Base may form the basis of all MARS model development.

At step 4215, pre-processing occurs. According to an embodiment of the invention, one of the first steps in the model development process is to study the data and its various characteristics. This process may ensure that adequate attention is given to the understanding of the problem space. Later, appropriate pre-processing steps may be taken to extract the maximum information out of the available data via a choice of a set of explanatory variables that have the maximum discriminatory power. According to an embodiment of the invention, as illustrated in Fig. 43, one of the early findings was the fact that for most of the candidate variables that were chosen on the basis of experience and judgment of the human underwriting experts the decision boundary regions as indicated by the human experts start at the tail-end of the variable distribution.

As described above, the decision problem may be to classify each applicant into risk classes, which are typically increasing in risk. Thus, as an example, the attribute denoted by the level of cholesterol in the blood of an individual may be considered. It is a known fact that a cholesterol level below 220 can be treated as almost normal. This suggests that in cases where the cholesterol level is at a certain level, such as up to about 240 at demarcation 4302, the human expert does not perceive a significant risk due to this factor. Thus, all cases with a cholesterol reading below this threshold can be grouped into a single class, *e.g.*, "Class 1," 4304 and the members in this class would not consequently impact the response variable (*e.g.*, the rate class decision).

As shown, a cholesterol level value of 240 is close to the 75th quantile 4306 of the distribution, while the value of 270 is in the 90th quantile range 4308.

One of the sub-steps may include location transformation and truncation 4216. A location transformation may be considered for all variables that exhibit the above property. Each variable may be transformed by subtracting out its normal value. This is realized by combining the knowledge of human experts as well, since for the majority of the attributes that are health related, there are well-documented and published normal thresholds.

According to an embodiment of the invention, it may not be desirable to differentiate among points within the normal ranges. Further, to focus the classifier on those in the abnormal range, the values of the variable may be saturated after a location transformation. In this case, the positive values may be considered, *e.g.*:

$$\text{NewValue} = \mathit{Max}(0, \text{OldValue} - \text{ReferenceValue})$$

The above is not a limitation of the general pre-processing step as would be applicable in other problems, but is a step relevant to the problem domain. There were variables which had the decision boundaries distributed fairly evenly over the entire range and did not warrant this specific transformation.

Further, another sub-step may include range normalization 4217. If it is desirable to compute distances in a multi-dimensional space, *e.g.*, to find the closest points to a given one, it may be necessary to normalize each dimension. Range normalization is typically the most common way to achieve this, *e.g.*:

$$\text{NewValue}\,\% = \frac{\text{NewValue} - \min_i(\text{NewValue}_i)}{\max_i(\text{NewValue}_i) - \min_i(\text{NewValue}_i)}$$

Another sub-step may involve "tag"-encoding 4218. According to an embodiment of the invention, a specialized set of variable encoding may also be used to extract the maximum information out of the decision space. This encoding may be referred to as the "tag." The tag is essentially an ordinal categorical variable developed from a collection of indicators for the various decision boundaries as defined by human experts. These indicators are evaluated for each relevant variable in the collection. The maximum of the individual indicators over the collection of variables results in the final "tag." For example, assume that there are four key variables (out of a larger number of fields in the case) that are highlighted by actuarial studies to determine mortality risk. Since the same studies indicate the critical thresholds that impact such risk, there is no reason to re-learn those thresholds. Therefore, they may be encoded in the indicator "tag." Table 5 below illustrates four variables: Nicotine History (NH), Body Mass Index (BMI), Cholesterol Ratio (Chol. Rat.), and Cholesterol Level (Chol. Lev.), and four groups of rules, one for each variable. According to this example, the value of the tag starts with a default of 1 and is modified by each applicable rule set. A running maximum of the tag value is returned at the end, as the final result of tag.

				TAG
A) Initialize:				1

B) Fire following rules

Rule #		IF		TAG
1	NH	<	t1	2
2		<	t2	3
3		<	t3	4
4	BMI	>	t4	2
5		>	t5	3
6		>	t6	4
7		>	t7	5
8		>	t8	6
9	Chol. Rat.	>	t9	2
10		>	t10	3
11		>	t11	4
12		>	t12	5
13		>	t13	6
14	Chol. Lev.	>	t14	2
15		>	t15	3
16		>	t16	4
17		>	t17	5
18		>	t18	6

C) Tag is determined by the MAX of the values determined by each of the four rule sets

Table 5

Thus, a tag may provide a utilization of the available human expert knowledge to obtain a boost in accuracy. By way of example, the models were built with and without the inclusion of the specialized "tag" variable and found that inclusion of the tag results in an improvement in accuracy by about 1-2% on average.

At step 4220, five-fold partitioning and resampling occurs, while a development and validation set is generated at step 4225. According to an embodiment of the invention, a stratified sampling methodology may be used to partition the data set into five equal parts. The stratification was done along the various rate classes to ensure a consistent representation in each partitioned sample. Further, a simple re-sampling technique may be used based on reusing each partition by taking out one part (done five times without replacement) as a holdout and recombining the remaining four and using it as a development sample to build a complete set of MARS models. This may be done five times, as mentioned earlier. By way of example, such a resampling and recombination was performed and the results were compared for consistency in accuracy, and also to note any fundamental shift in models. The accuracy measures were found to be closely grouped in the 94.5%-95.5% neighborhood and with model consistency throughout.

At step 4230, model-building experiments are performed. According to an embodiment of the invention, a variety of exploratory regression models may be built and trained on the CB development sets. Further, their classification accuracy may be tested and validated on the CB validation sets. According to an embodiment of the invention, a parallel-network of MARS models may evolve and develop from a global regression model and a classification and regression trees ("CART") model, and allows the use of MARS in the framework of a multi-class classification problem. The global regression model and the classification and regression trees ("CART") model will now be described in greater detail below.

Since this is a multi-class classification problem, by definition the response variable is a polychotomous categorical variable, *i.e.*, a variable that can take values from a set of labels (*e.g.*, "Preferred Best," "Preferred," "Select," "Standard Plus," "Standard"). However, since in this case the response is ordinal (the order of the categorical values reflects the corresponding increasing risk), a risk metric may be obtained such as from an actuarial department of the insurance company. This allows the mapping of the categorical values to numerical values (*e.g.*, reflecting mortality risk) and treating the response variable as a continuous one in order to fit a global multivariate linear regression. Using this method, a moderate fit to the data is obtained. However, the maximum accuracy achieved was about 60%, far from the desired accuracy level of above 90%.

Additionally, a CART based model may be built using the data. To maintain robustness and to avoid the possibility of overfitting the model, it may be necessary to minimize the structural complexity of the CART model. This approach yielded a CART tree with about 30 terminal nodes. Its corresponding accuracy level was substantially better than the global regression and was about 85%. Increasing the accuracy for the training sets would have resulted in deeper, more complex trees, with larger number of terminal nodes. Such trees would exhibit overfitting tendencies and poor generalization capabilities, leading to low accuracy and robustness when evaluated on the validation sets.

From these experiments, it can be determined that a global regression model, which is essentially a main-effects fit, has moderate explanatory power, but a CART tree, which is a local non-parametric model, has a much better performance. Since CART is essentially a pure interaction-based model the motivation for a MARS based modeling schema was obvious, as MARS allows both main and interaction effects to be incorporated into the model, and being a piecewise-linear adaptive regression procedure, MARS can approximate very well any non-linear structure (if present). Since the original motivation of development of the MARS algorithm stemmed from the problem of discontinuity of CART terminal node estimates, the same benefits may apply here.

At step 4235, a parallel-network of MARS models is implemented. According to an embodiment of the invention, one issue involved the difficulty of global models to incorporate the jumps in decision boundaries of majority of the variables in an extremely small bounded range. In other words, since the decision boundaries begin only after the 75th quantile value of the explanatory variable, the shift over all other decision variables usually occur by the 95th quantile. This issue may be addressed in a number of ways. According to one approach, "tag" encoding as explained above helps the MARS search algorithm to find the "knots" in the right place.

According to another approach, a "parallel network" arrangement of models may be used. A parallel network arrangement is a collection of MARS models, each of which solves a binary, or two-class problem. This may take advantage of the fact that the response variable is ordinal *e.g.*, the decision classes being risk categories are increasing in risk. The approaches to these issues should not be considered as limitations of the methodology presented here, but rather a property explored in order to achieve better results. In addition, the above case generalizes to handle problems where the response may not be ordinal.

An advantage of the order of the response variable may be taken by building two models each for every rate class, except the boundary classes, with one model for each side. For easier reference, the two models may be referred to as the left model and the right model. Fig. 44 illustrates an example of such models. A population 4402 is divided into non-smoking applications 4404, non-underwritten applications 4406 and nicotine applications 4408. The "Preferred" class has been broken down into a "Preferred Left" model 4410 and "Preferred Right" model 4412. The minimum of the two models is selected, *e.g.*, M(Pref) = min (L,R), 2814. The results are then input into the aggregation module 4416, which aggregates all results from the binary classifiers and selects the rate class that best fits a given application. For example, for the rate class "Preferred," two models are built which estimate class membership value. The "Left" model distinguishes all preferred cases from cases of classes, which are to the left of preferred while the "Right" model does the opposite. The final class membership value may be the minimum of these two membership values obtained. Further, in the general case where there is no known order amongst classes, the Left/Right models may collapse into a single model providing with one estimated membership value.

According to an embodiment of the invention, the MARS methodology may be adapted to handle logistic regression problems in the classical sense. Such an adaptation would need an adjustment of the lack-of-fit ("LOF") criteria to be changed from least squares to logistic. However, logistic regression procedure is in itself a likelihood maximization problem that is typically solved by using an iteratively re-weighted least squares ("IRLS") algorithm or its counterparts. The viability of MARS may depend on the fast update criteria of the least squares LOF function, which an IRLS logistic estimation would generally prohibit.

According to an embodiment of the invention, an approximation may be made to use the final set of MARS variables back into a SAS logistic routine and refit. As said before, this is an approximation because if one could ideally use logistic LOF function, then one could have derived the optimal set of logistic candidate variable transforms. However, a re-fit process may still achieve the same degree of fit and provide model parsimony in some of the subset models built. Also, since the logistic function is a (0,1) map, this gives class membership values that can be treated as probabilities.

According to an embodiment of the invention, a MARS module may be implemented with software code in SAS and using MARS, where the code has been trained and tested using the five-fold partitions method described above. By way of example of the results of such an implementation, Fig. 45 illustrates a comparison matrix 4500 (with a dimensionality of *k* x *k*), whose *k* columns contain the set of possible decisions available to the classifier, and whose *k* rows contain the correct corresponding standard reference decision, can describe a classifier's performance on a given data set, is illustrated in Fig. 45.

In this example, agreement between the classifier and the standard reference decision occurs when the case results on the main diagonal of matrix 4500 while any other cell above or below the main diagonal contains misclassified cases. In the illustrative example depicted in Fig. 45, for the second row of 4502, labeled "Preferred," 360 out of total of 374 cases were correctly assigned to that rate class, while 1 was assigned to "P Best," 11 to "Select", 1 to "Standard" and 1 to "Send to Underwriter."

As shown in Fig. 46, 4602 refers to the total number of agreements between the classifier and the standard reference decisions for non-smokers, while 4608 refers to the total number of agreements between the classifier and the standard reference decisions for smokers. The notations 4604 and 4606 refer to the total number of disagreements between the classifier and the standard reference decisions for non-smokers, while 4610 and 4612 refer to the total number of disagreements between the classifier and the standard reference decisions for smokers. 4614 refers to the total number of agreements not to make a decision and send the case to UW (*e.g.*, underwriter) and notations 4616 and 4618 refer to the total number of disagreements not to make a decision and send to UW.

Further, the matrix depicted in Fig. 46 may be used to illustrate the performance measures used in the evaluation of the classifiers. Let *N* be the total number of cases considered (in this example, *N*= 2,920). According to the annotation in Fig. 46, N = m1 + m2 + m3 + m4 + m5 + m6 + m7 + m8 + m9. In this example, *N2* = 182, which is the sum of all cases that should have been sent to the human underwriter (*i.e.*, m9+ m7 in Fig. 46), and therefore *N1* = (2,920-182) = 2,738. Three measures of performance for the classifier may be used, where *M(i,j)* is a cell in the matrix shown in Fig. 45:

- Coverage: the total number of decisions made by the classifier as a percentage of the total number of cases considered, *i.e.*:

$$\boxed{\text{Coverage}=\sum_{i=1}^{k}\sum_{j=1}^{k-1}M(i,j)/N}$$

Using the annotations defined in Fig. 46, coverage may be redefined as:

$$Coverage=\frac{(m1+m2+m3)+(m4+m5+m6)+m9}{(m1+m2+m3)+(m4+m5+m6)+(m7+m8+m9)}$$

Thus, in the example depicted in Fig. 45 the coverage is: (2,920-242)/2,920 = 91.71%. An addition performance measure may include:

- Relative Accuracy: the total number of correct decisions made by the classifier as a percentage of the total number of decisions made, *i.e.*:

$$\boxed{\text{Relative Accuracy}=\sum_{i=1}^{k-1}M(i,i)/\sum_{i=1}^{k}\sum_{j=1}^{k-1}M(i,j)}$$

Using the annotations defined in Fig. 46, the relative accuracy may be redefined as:

$$RelativeAccuracy = \frac{m1 + m4}{(m1 + m2 + m3) + (m4 + m5 + m6) + m9}$$

In the example depicted in Fig. 45 the relative accuracy is: (2,558)/(2,920 –242)= 95.52%. An further performance measure may include:

- Global Accuracy: the total number of correct decisions made by the classifier as a percentage of the total number of cases considered, *i.e.*:

$$\boxed{\text{Global Accuracy} = \sum_{i=1}^{k} M(i,i) / \sum_{i=1}^{k}\sum_{j=1}^{k} M(i,j)}$$

Again, using the annotations defined in Fig. 46, the global accuracy may be redefined as:

$$GlobalAccuracy = \frac{m1 + m4 + m7}{(m1 + m2 + m3) + (m4 + m5 + m6) + (m7 + m8 + m9)}$$

In the example depicted in Fig. 45 the global accuracy is: 2,734/2,920= 93.63%. Coverage and relative accuracy may be competing objectives. By establishing a confidence metric for the classifier output, one could adjust a confidence threshold to achieve various tradeoffs between accuracy and coverage. At one extreme, one could have a very low tradeoff, accepting any output (this would yield 100% coverage but very low accuracy). At the other extreme, one could have very high confidence thresholds. This would drastically reduce coverage but increase relative accuracy.

The results of networks of MARS (or Neural Networks, as described below) models could also be post-processed to establish an alternative confidence metric that could be used to achieve other tradeoffs between accuracy and coverage. The tables set forth in Fig. 47 describe the performance of the network of MARS models on each of the five partitions. For each partition, the global and relative accuracy is listed, with the corresponding coverage. The results are shown with and without post-processing.

Each of these partitions (*e.g.*, Partition 1, 4710, Partition 2, 4720, Partition 3, 4730, Partition 4, 4740 and Partition 5, 4750) shows the performance results of the network of MARS models applied to 80% of the data used to build the model (training set 4760) and 20% of the data that was withheld from the model construction (validation set 4770). The tables in Fig. 48 summarize the minimum 4810, maximum 4820, and average 4830 results of applying the network of MARS models to the five partitions.

These tables illustrate that the average performance of a network of MARS models, applied to the five partitions, was very accurate. In particular a relative accuracy of 95% on the validation set 4840 of Fig. 48, with coverage of about 90% may be extremely good and useful for quality assurance. An analysis of the minimum and maximum achieved may also show a high level of robustness, exemplified by the relatively tight range of performance values.

The technical considerations that go into a MARS model are well known and can be found in Friedman's original paper in the *Annals of Statistics*, the contents of which are incorporated herein by reference. However, to better illustrate the present invention, it is useful to describe a few basic points adopted in the MARS tuning as well as some additional steps that may be necessary to ensure a robust model building process.

General MARS parameters may include overfit and cost-complexity pruning, cross-validation, and multi-collinearity. According to an embodiment of the invention, MARS is essentially a recursive-partitioning procedure. The partitioning is done at points of the various explanatory variables defined as "knots" and overall optimization is achieved by performing knot optimization over the lack-of-fit criteria. Moreover, to achieve continuity across partitions MARS employs a two-sided power basis function of the form:

$$b_q^{\pm}(x-t) = [\pm(x-t)]_+^q.$$

However, in this case, a linear-piecewise basis q=1 is used. Here 't' is the knot around which the basis is formed. It may be important to use an optimal number of basis functions to guard against possible overfit. By way of example, an experiment may be performed with one dataset by starting from a small number of maximal basis functions and building it up to a medium size number and use the cost-complexity notion developed in CART methodology and deployed in MARS to prune back and find a balance in terms of optimality which provides an adequate fit. In this example, the use of cost-complexity pruning revealed that 25-30 basis functions were sufficient.

Another important criteria which affects the pruning is the estimated degrees of freedom allowed. This may be done by using ten-fold cross validation from the data set for each model.

In addition, there is no explicit way by which MARS can handle multi-collinearity. However, MARS does provide a parameter that penalizes the separate choice of correlated variables in a downstream partition. MARS then works with the original parent instead of choosing other alternates. According to an embodiment of the invention, a medium penalty may be used to take care of this problem.

Further, optimization of cut-offs using evolutionary algorithms ("EA") may be used. When a new case comes in, it is evaluated by the complete set of models and a class membership distribution is obtained for every incoming case. Next in line comes the problem of assigning rate-classes to the incoming case. One alternative may be to use hand-tuned cut-offs computed through simple tools like Microsoft Excel based solver. These results may be compared to an EA based optimized cut-off set. By way of example, an evolutionary algorithm may provide a boost in accuracy by about 1% as compared to the hand-tuned cut-offs.

5. Neural Network Classifier

Another aspect of the present invention may provide a method and system to implement a neural network classifier with multiple classes for automated insurance underwriting and its quality assurance. Neural networks may be advantageous, as they can approximate any complex nonlinear function with arbitrary accuracy (*e.g.*, they are universal functional approximators). Neural networks are generally non-parametric and data-driven. That is, they approximate the underlying nonlinear relationship through learning from examples with few a priori assumptions about the model. In addition, neural networks are able to provide estimates of posterior probabilities. Such posterior probability values may be useful for obtaining the highest possible decision accuracy in the classifier fusion or other decision-making processes.

There are a variety of types of neural networks. However, neural networks can be broadly categorized into two main classes, *i.e.*, feed-forward and recurrent (also called feed back) neural networks. Among all these types, multiple-layer feed-forward neural networks are often used for classification. Neural networks can be directly applied to solve both dichotomous and polychotomous classification problems.

However, it is generally more accurate and efficient when neural networks are used for two-class (*e.g.*, dichotomous) classification problems. As the number of classes increases, direct use of multi-class neural networks may encounter difficulties in training and in achieving the desired performance.

As previously described, insurance underwriting problems may often involve the use of large numbers of features in the decision-making process. The features typically include the physical conditions, medical information, and family history of the applicant. Further, insurance underwriting frequently has a large number of risk categories (*e.g.*, rate classes). The risk category of an application is traditionally determined by using a number of rules/standards, which often have the form of "if the value of feature *x* exceeds *a*, then the application can't be rate class *C*, *i.e.*, has to be lower than *C*". These types of decision rules, 4930 and 4940 in Fig 49, "clip" the decision surface. Decision rules interpreted and used by a human underwriter may form an overall piecewise-continuous decision boundary, as shown in the graph of Fig. 49.

To design a neural network classifier to achieve a comparable performance (*e.g.*, accuracy and coverage) as rule-based classifiers for insurance underwriting, various issues may need to be addressed. First, a neural network may need to deal with a large number of features and target classes. The large number of features and high number of target classes call for a high degree of complexity of neural network ("NN") structure (*e.g.*, more nodes and more parameters to learn, i.e. higher Degrees of Freedom (DOF). Such complex NN structures may require more training data for properly training the network and achieving reasonable generality (performance). However, sufficient data may be difficult to obtain. Even with sufficient data, the complex neural network structure requires enormous training time and computational resources. More importantly, complex NN structures (high DOF) tend to have more local minima, and thus, training is prone to fall into local minima and fails to achieve global minimization. As a result, it usually difficult to achieve a desired performance for a neural network with complex structure.

Another issue to be addressed involves incorporating domain knowledge into the neural network classification process. As discussed before, the discrete rules that human underwriters use for risk category assignment form an overall piecewise-continuous decision boundary in the feature space and neural networks may have difficulty learning the decision boundary due to the insufficient data points being available. One way to alleviate the difficulty and improve the performance of the neural network may be to directly incorporate the rules into the neural network model and use these rules as additional information to "guide" network learning.

One aspect of the present invention is related to a method and system of improving the performance of neural network classifiers, so that the neural network classifier can perform automated insurance underwriting and its quality assurance with a level of accuracy and reliability that is comparable to the rule-based production decision engine. Specifically, this invention improves the performance of classifiers by decomposing a multi-class classification problem into a series of binary classification problems. Each of the binary classifiers may classify one individual class from the other classes and the final class assignment for an unknown input will be decided based on the outputs of all of the individual binary classifiers.

Additionally, as another way to improve the classifier performance, this invention incorporates the domain knowledge of the human underwriter into a neural network design. The domain knowledge, represented by a number of rules, may be integrated into a classifier by using an auxiliary feature, the value of which is determined by the rules. Moreover, to further improve the classifier performance, this invention may also analyze the outputs of the individual binary classifiers to identify the difficult cases for which the classifier cannot make a solid decision. To reduce misclassification rate, these difficult cases may then be sent to a human underwriter for further analysis.

In the conventional design of multi-class neural network classifiers. a single neural network contains multiple output nodes. According to an embodiment of the invention, decomposing the multi-class classifier into multiple binary classifiers may solve a multi-class classification problem. For the purposes of illustration, assume that a hypothetical life insurance company has risk categories "Cat1", "Cat2", "Cat3", "Cat4", and "Cat5". A rating of "Cat1" is the best risk, while "Cat5" is the worst. Then, the concept of the multi-class classifier decomposition used in this invention can be illustrated in the example of Fig. 50. Each binary classifier (5010, 5020, 5030) is for one class and is trained to classify the specific class (the "class") and the rest of the classes combined (the "others'). Before training each of the binary classifiers, the training set is relabeled "1" for the data points in the "class" group and "0" for the data points in the "others" group. When performing classification for a new input case, each of the binary classifiers determines the probability that the new case belongs to the class for which the binary classifier is responsible. Therefore, the output of the neural network is a number in the [0,1] interval. The final class for the new input case is assigned by the MAX decision rule 5040. For example, an application may receive a "0.6 and a 1" in the Cat3 and Cat4 categories, respectively, and a "0" in the Cat1, Cat2, and Cat5 risk categories. The MAX decision rule 5040 may then select the Cat4 risk category.

According to an embodiment of the invention, for each of the binary classifiers designed in the current invention, the neural network is multiple-layer feed-forward in type and has one hidden layer. However, for other applications, using different neural network types with more than one hidden layer may be explored for obtaining better performance. It is therefore to be understood that the current invention is not limited to one hidden layer feed-forward neural networks. Instead, the method may work equally well for multiple numbers of hidden layers.

According to an embodiment of the invention, domain knowledge may be integrated into neural network learning by representing the knowledge with an auxiliary feature. The domain knowledge may be first represented by a series of rules. A typical rule has the following format (once again using the afore-mentioned five hypothetical rate classes): "If the applicant's cholesterol level exceeds 252, he does not qualify for rate class C1, *i.e.*, the best rate class for him is C2". Formally, this rule can be expressed in a general IF-THEN rule as follows.

IF $x_i > t_{i,j}$, **THEN** the best available rate class is C_j

where x_i is the i^{th} feature, $t_{i,j}$ is the j^{th} threshold of the i^{th} feature, and C_j is the j^{th} rate class. The incorporation of domain knowledge is further described below.

According to an embodiment of the invention, the classifier design process for a neural network classifier may comprise data preprocessing, classifier design and optimization, and post-processing. These three aspects are described in greater detail below.

Data preprocessing may include range normalization and feature extraction and selection. According to an embodiment of the invention, range normalization is a process of mapping data from the original range to a new range. Normalization may be generally problem specific. However, it is often done either for convenience or for satisfying the input requirements of the algorithm(s) under consideration. For pattern classification problems, one purpose of normalization is to scale all features the classifier is using to a common range so that effects due to arbitrary feature representation (*e.g.*, different units) can be eliminated. In addition, some classifiers, such as neural networks, require a range of input to be normalized.

One way to normalize data is range normalization. To normalize the data by range, the feature value is divided by its range, *i.e.*, the difference between the maximum and the minimum of the feature value. Let $x_{i,j}$ be value of the i^{th} data point of the j^{th} feature.

Then the normalized value $y_{i,j}$ is:

$$y_{i,j} = \frac{x_{i,j} - \min(x_{,j})}{\max(x_{,j}) - \min(x_{,j})} \qquad (1)$$

The normalized values $y_{i,j}$ will be in the range of [0, 1]. The range normalization requires knowing the minimum and the maximum values of the data. The greatest advantage of this normalization is that it introduces no distortion to the variable distribution, as the instance values and their corresponding normalized values have a linear relationship. That is, given two instance values with the first being twice the second, when they are normalized the first normalized value will still be twice the second normalized value. This is why range normalization is also called linear scaling or linear transformation.

Another type of data preprocessing may involve feature extraction/selection. For example, raw data is placed within a 20-column spreadsheet. The first column is the applicant ID number and the second column is the rate class. Columns 3 through 20 are the attributes/variables/features for the applicant. Instead of directly using the 18 original features, two new features are derived. The first derived feature is the body mass index ("BMI"). Underwriter experience has shown that the BMI has more discriminating power in classification. The second derived feature, *tag*, is used to represent the domain knowledge in neural network training. The two derived features are further described below.

As described above, BMI is defined as ratio of weight in kilogram and the height squared in meters. Let *wt* be the weight in pounds and *Ht* be the height in inches. BMI can be expressed as:

$$BMI = \frac{wt \times 2.2046}{[Ht \times 0.0254]^2} \tag{2}$$

One approach for incorporating domain knowledge into the neural network modeling involves training by hints, as described by Abu-Mostafa (1993), where almost any type of prior knowledge can be incorporated into a neural network through constructing the hints. Although the technique is flexible, it may be of a limited application in neural networks. According to an embodiment of the invention, domain knowledge is incorporated into the neural network classifier by using an artificial feature, such as tag. The tag feature may take different values based on a set of rules that represent the domain knowledge.

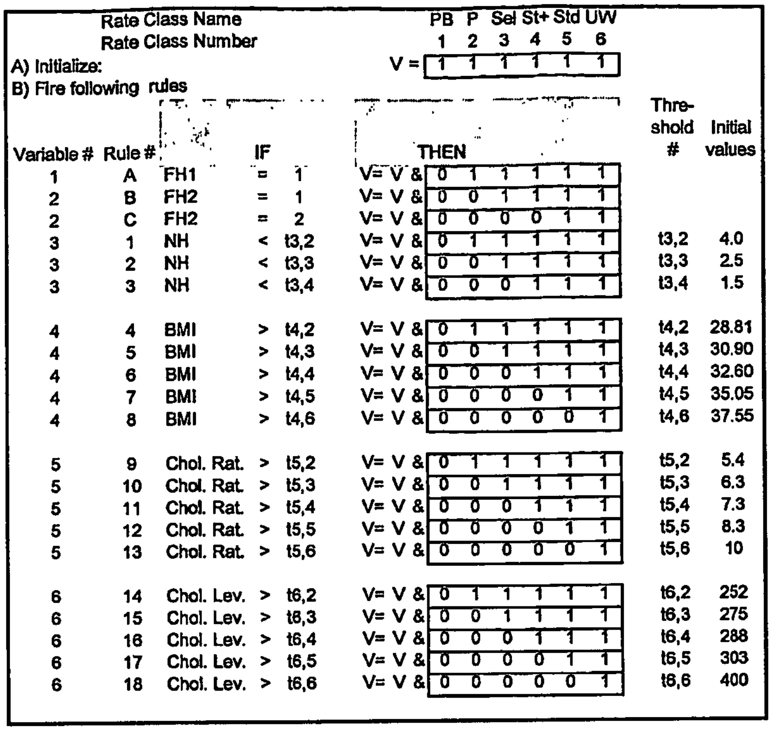
By way of example, the five family history features, such as from columns 3-7, are condensed and represented by two features, FH1 and FH2. While the FH1 feature has the binary values of 0 or 1, FH2 has the triple values of 0,1, and 2. The values of FH1 and FH2 are determined by the following rules, where the terms *age_sib_card_canc_diag, age_moth_card_canc_diag, age_fath_card_canc_diag, age_moth_card_death, age_fath_card_death* respectively correspond to the age when a sibling of the applicant was diagnosed with a cardiac or cancer disease, the age when the mother of the applicant was diagnosed with a cardiac or cancer disease, the age when the father of the applicant was diagnosed with a cardiac or cancer disease, the age when the mother of the applicant died due to a cardiac disease, and the age when the father of the applicant died due to a cardiac disease. For a given applicant, one or more of these terms may be not applicable.

```
IF (age_sib_card_canc_diag $\leq 60$)
        OR (age_moth_card_canc_diag $\leq 60$)
        OR (age_fath_card_canc_diag $\leq 60$),
THEN $FH_1$ is 1.
Otherwise, $FH_1$ is 0.


IF (age_moth_card_death $\leq 60$) OR (age_fath_card_death $\leq 60$),
THEN $FH_2=1$.


IF (age_moth_card_death $\leq 60$) AND (age_fath_card_death $\leq 60$),
THEN $FH_2=2$.
Otherwise, $FH_2=0$.
```

Examples of rules that may be used to compute TAG are listed below in Table 6.



	Rate Class Name					PB	P	Sel	St+	Std	UW		
	Rate Class Number					1	2	3	4	5	6		
A) Initialize:					V =	1	1	1	1	1	1		
B) Fire following rules													
Variable #	Rule #	IF			THEN							Thre-shold #	Initial values
1	A	FH1	=	1	V= V &	0	1	1	1	1	1		
2	B	FH2	=	1	V= V &	0	0	1	1	1	1		
2	C	FH2	=	2	V= V &	0	0	0	0	1	1		
3	1	NH	<	t3,2	V= V &	0	1	1	1	1	1	t3,2	4.0
3	2	NH	<	t3,3	V= V &	0	0	1	1	1	1	t3,3	2.5
3	3	NH	<	t3,4	V= V &	0	0	0	1	1	1	t3,4	1.5
4	4	BMI	>	t4,2	V= V &	0	1	1	1	1	1	t4,2	28.81
4	5	BMI	>	t4,3	V= V &	0	0	1	1	1	1	t4,3	30.90
4	6	BMI	>	t4,4	V= V &	0	0	0	1	1	1	t4,4	32.60
4	7	BMI	>	t4,5	V= V &	0	0	0	0	1	1	t4,5	35.05
4	8	BMI	>	t4,6	V= V &	0	0	0	0	0	1	t4,6	37.55
5	9	Chol. Rat.	>	t5,2	V= V &	0	1	1	1	1	1	t5,2	5.4
5	10	Chol. Rat.	>	t5,3	V= V &	0	0	1	1	1	1	t5,3	6.3
5	11	Chol. Rat.	>	t5,4	V= V &	0	0	0	1	1	1	t5,4	7.3
5	12	Chol. Rat.	>	t5,5	V= V &	0	0	0	0	1	1	t5,5	8.3
5	13	Chol. Rat.	>	t5,6	V= V &	0	0	0	0	0	1	t5,6	10
6	14	Chol. Lev.	>	t6,2	V= V &	0	1	1	1	1	1	t6,2	252
6	15	Chol. Lev.	>	t6,3	V= V &	0	0	1	1	1	1	t6,3	275
6	16	Chol. Lev.	>	t6,4	V= V &	0	0	0	1	1	1	t6,4	288
6	17	Chol. Lev.	>	t6,5	V= V &	0	0	0	0	1	1	t6,5	303
6	18	Chol. Lev.	>	t6,6	V= V &	0	0	0	0	0	1	t6,6	400

Table 6

As indicated earlier, domain knowledge may be represented by a set of rules. A typical rule may have the following format (once again using the afore-mentioned five hypothetical rate classes): "If the applicant's cholesterol level exceeds 252, he does not qualify for rate class C1, *i.e.*, the best rate class for him is C2". For example, this rule can be expressed in a general IF-THEN rule as follows:

IF $x_i > t_{i,j}$, **THEN** the best available rate class is C_j

Where, x_i is the i^{th} feature, $t_{i,j}$ is the j^{th} threshold of the i^{th} feature, and C_j is the j^{th} rate class.

A vector with binary number “0” or “1” may be used to represent the consequent part of the IF-THEN rule. For example, [0, 1, 1, 1, 1] means the best rate class of C_2 while [0, 0, 0, 1, 1] means the best rate class of C_4.

For each data point in the training data set, all rules that “fire” are checked and the intersection (*e.g.*, the Boolean logic minimum) of the vector of the firing rule is calculated, as well as the vector that has initial value of all ones. The value of the auxiliary feature may then be determined by counting the number of ones in the final vector. As can be seen, the auxiliary feature takes integer numbers ranging from one to the number of rate classes. The pseudo-code shown summarizes the procedure of determining the value of the auxiliary feature.

```
FOR each of the data points in the training set
    Initialize vector V=[1, 1, 1, 1, 1]
    FOR each of the rules
        IF the $i^{th}$ rule is fired, THEN V=V & Vi (“&” is logic AND)
    END of all rules
    The value of the auxiliary feature = the number of ones in the vector V.
END of all data points
```

After obtaining the value of the auxiliary feature for each data point, the auxiliary feature may be treated as a regular feature and included into the final feature set. The neural network may then be trained and tested with the final feature set. Because of the additional information provided by the auxiliary feature, the neural network may be “guided” during learning to more quickly find the piecewise continuous decision boundary, which not only reduces the training time and efforts, but may also improve the classification performance of neural network classifier.

Additional features that may be used for neural network classifier design include, but are not limited to, tag, BMI, diastolic and/or systolic blood pressure readings, cholesterol level, cholesterol ratio, various liver enzymes, such as SGOT (Serum Glutamic Oxaloacetic Transaminase), SGPT (Serum Glutamic Pyruvic Transaminase), GGT (Galactan Galactosyl Transferase), nicotine use history, and various aspects of family history.

There are a number of types of neural networks. According to an embodiment of the invention, a three-layer feed-forward neural network with back propagation learning may be used. Two separate models may be used for nicotine and non-nicotine cases, respectively. By way of example, for nicotine cases, there may be three rate classes, *e.g.*, "Preferred_nic," "Standardplus_nic," and Standard_nic, while non-nicotine cases may have five rate classes, *e.g.*, "Best," "Preferred," "Select," "Standardplus," and "Standard." Both models are multiple-class classifiers. A neural network with multiple output nodes may be a typical design for multiple-class classifiers where each of the neutral network output nodes corresponds to each class. However, neural networks with multiple output nodes may have a large number of weights and biases, and thus require a large training data set and more training time for properly training the network. If the data size is relatively small compared to the number of features and the number of classes, multiple binary neural networks may be used to perform the multiple-class classification. Using multiple binary-networks may reduce the complexity of the network, thus reducing the training time, but also may improve the classification performance. An example of the architecture of a neural network classifier is illustrated in Fig. 51. The non-nicotine model 5110 has five binary classifiers 5120 while the nicotine model 5130 has three binary classifiers 5140. Each model 5110, 5130 has a MAX function 5150 and 5160. Applications in the non-nicotine model 5110 are then assigned to the appropriate rate class 5170, while applications in the nicotine model 5130 are assigned to the appropriate rate class 5180.

In the example of Fig. 51, each binary network has the structure of 12-5-1, *e.g.*, twelve input nodes, five hidden neurons, and one output node. Activation functions for both hidden and output neurons may be logistic sigmoidal functions. According to an embodiment of the invention, the range of target values may scaled to [0.1 0.9] to prevent saturation during training process. The Levenberg-Marquardt numerical optimization technique may be used as the backpropagation-learning algorithm to achieve second-order training speed.

Each binary network represents an individual rate class and is trained with the targets of one-vs-other. During classification for an unknown case, each network provides the probability of the unknown case belonging to the class it represents. The final rate class of the unknown case is determined by the MAX decision rule, *e.g.*, given a vector whose entry values are in the interval [0,1], the MAX rule will return the value of the position of the largest entry.

To further improve the classification performance, it may be advantageous to apply some post-processing techniques to the outputs of the individual networks, prior to the MAX decision making process. Instead of assigning rate class to an unknown case just based on the maximum outputs of the individual networks, the distribution of the outputs is characterized. If the distribution of the outputs does not meet certain pre-defined criteria, no decision needs to be made by the classifier. Rather, the case will be sent to human underwriter for evaluation. The rationale here is that if a correct decision cannot be made, it would be preferable that the classifier makes no decision rather than the wrong decision. Considering the neutral network outputs as discrete membership grade for all rate classes, the four features that characterize the membership grades may be the same as those set forth above with respect to the fusion module discussed above, *i.e.*, cardinality, entropy, the difference between the highest and the second high values of outputs, and the separation between rank orders of the highest and the second highest values of outputs.

Again, with the features defined for characterizing the network outputs, the following two-step criteria may be used for "rejecting" the cases:

$$\text{Step 1: } C < \tau_1 \text{ OR } C > \tau_2 \text{ OR } E > \tau_3$$

$$\text{Step 2: } D < \tau_4 \text{ AND } S \leq 1$$

Where τ_1, τ_2, τ_3, and τ_4 are the thresholds. The value of the thresholds is typically data set dependent. In this embodiment, the value of the thresholds are first empirically estimated and then fine-tuned by evolutionary algorithms (EA). The final numbers for all five-fold data sets are illustrated in Table 7 below:

Non-nicotine model					
	Run#1	Run#2	Run#3	Run#4	Run#5
τ_1	0.5	0.5	0.5	0.5	0.5
τ_2	2.0	2.0	2.0	2.0	2.0
τ_3	0.9	0.9	0.9	0.93	0.98
τ_4	0.1	0.15	0.1	0.1	0.07
Nicotine model					
	Run#1	Run#2	Run#3	Run#4	Run#5
τ_1	0.3	0.3	0.3	0.3	0.3
τ_2	1.75	1.75	1.75	1.75	1.75
τ_3	0.85	0.85	0.8	0.85	0.85
τ_4	0.2	0.25	0.2	0.2	0.2

Table 7

According to an embodiment of the invention, a neural network classifier may be implemented using software code, and tested against a case base. By way of example, a software implementation of a neural network may use a case base of 2,879 cases. After removal of 173 UW cases, the remaining 2,706 cases were used for training and testing the neural network classifier. Five-fold cross-validation was used to estimate the performance of the classifier.

The combined confusion matrices of the five-fold runs are illustrated in Fig. 52. For comparison, the combined confusion matrices for the five-fold runs after post-processing are illustrated in Fig. 53. The performance for this example before post-processing is provided in Fig. 54, while the performance for this example after post-processing is provided in Fig. 55.

According to an embodiment of the invention, the systems and processes described in this invention may be implemented on any general purpose computational device, either as a standalone application or applications, or even across several general purpose computational devices connected over a network and as a group operating in a client-server mode. According to another embodiment of the invention, a computer-usable and writeable medium having a plurality of computer readable program code stored therein may be provided for practicing the process of the present invention. The process and system of the present invention may be implemented within a variety of operating systems, such as a Windows® operating system, various versions of a Unix-based operating system (*e.g.*, a Hewlett Packard, a Red Hat, or a Linux version of a Unix-based operating system), or various versions of an AS/400-based operating system. For example, the computer-usable and writeable medium may be comprised of a CD ROM, a floppy disk, a hard disk, or any other computer-usable medium. One or more of the components of the system or systems embodying the present invention may comprise computer readable program code in the form of functional instructions stored in the computer-usable medium such that when the computer-usable medium is installed on the system or systems, those components cause the system to perform the functions described. The computer readable program code for the present invention may also be bundled with other computer readable program software. Also, only some of the components may be provided in computer-readable code.

Additionally, various entities and combinations of entities may employ a computer to implement the components performing the above-described functions. According to an embodiment of the invention, the computer may be a standard computer comprising an input device, an output device, a processor device, and a data storage device. According to other embodiments of the invention, various components may be computers in different departments within the same corporation or entity. Other computer configurations may also be used. According to another embodiment of the invention, various components may be separate entities such as corporations or limited liability companies. Other embodiments, in compliance with applicable laws and regulations, may also be used.

According to one specific embodiment of the present invention, the system may comprise components of a software system. The system may operate on a network and may be connected to other systems sharing a common database. Other hardware arrangements may also be provided.

Other embodiments, uses and advantages of the present invention will be apparent to those skilled in the art from consideration of the specification and practice of the invention disclosed herein. The specification and examples should be considered exemplary only. The intended scope of the invention is only limited by the claims appended hereto.

While the invention has been particularly shown and described within the framework of an insurance underwriting application, it will be appreciated that variations and modifications can be effected by a person of ordinary skill in the art without departing from the scope of the invention. For example, one of ordinary skill in the art will recognize that certain classifiers can be applied to any other transaction-oriented process in which underlying risk estimation is required to determine the price structure (*e.g.*, premium, price, commission, *etc.*) of an offered product, such as insurance, re-insurance, annuities, *etc.* Furthermore, one of ordinary skill in the art will recognize that such decision engines do not need to be restricted to insurance underwriting applications.

CLAIMS

What is claimed is:

1. A system for underwriting an insurance application based on a plurality of previous insurance application underwriting decisions, where underwriting the insurance application includes assigning a classification to the insurance application, the system comprising:

a processing module (116), where the processing module:

a) processes a plurality of features associated with the insurance application; and

b) outputs a processed result for each of the plurality of features;

a plurality of classifiers (5010, 5020, 5030), where there is a classifier for each classification and where each of the classifiers:

a) receives the plurality of processed results; and

b) outputs a classification result for the insurance application;

at least one post-processor module (1026), where the at least one post-processing module:

a) processes each of the classification results; and

b) outputs a processed classification result; and

a classification selector (5040) for selecting one of the processed classification results, where the selection indicates a classification assignment for the insurance application and selects the highest of the processed classification results.

2. The system according to claim 1, where the plurality of classifiers are binary classifiers.

3. The system according to claim 2, where the classification results from each of the plurality of binary classifiers (5010, 5020, 5030) comprise one of:

a) the selection of the classification associated with the binary classifier; and

b) a selection of the rest of the classifications associated with the other binary classifiers.

4. The system according to claim 1, where the processing performed by the pre-processing module comprises at least one of:

a) range normalization; and

b) feature extraction.

5. The system according to claim 1, where the processing performed by the pre-processing module comprises generating a tag, wherein the tag is generated based on one or more rules applied to at least one of the plurality of features.

6. The system according to claim 1, where the processing performed by the post-processing module (1026) comprises at least one of:

a) determining cardinality for the plurality of classification results;

b) determining entropy for the plurality of classification results;

c) determining the difference between the highest and second highest values of the classification results; and

d) determining the separation between the rank orders of the highest and second highest values of the classification results.

7. The system according to claim 1, where the selector module (5040) is a maximum classification selector which selects the highest of the processed classification results.

8. A system for underwriting an insurance application based on a plurality of previous insurance application underwriting decisions, where underwriting the insurance application includes assigning a classification to the insurance application, the system comprising:

a processing module (5040), where the processing module:

a) processes a plurality of features associated with the insurance application, where the processing performed by the pre-processing module comprises at least one of:

i) range normalization; and

ii) generating a tag, wherein the tag is generated based on one or more rules applied to at least one of the plurality of features; and

b) outputs a processed result for each of the plurality of features;

a plurality of binary classifiers (5010, 5020, 5030), where there is a binary classifier for each classification and where each of the binary classifiers:

a) receives the plurality of processed results; and

b) outputs a classification result for the insurance application;

at least one post-processor module (1026), where the at least one post-processing module:

a) processes each of the classification results; and

b) outputs a processed classification result; and a classification selector (5040) for selecting one of the processed classification results, where the selection indicates a classification assignment for the insurance application and selects the highest of the processed classification results.

9. The system according to claim 8, where the classification results from each of the plurality of binary classifiers (5010, 5020, 5030) comprise one of:

a) the selection of the classification associated with the binary classifier; and

b) a selection of the rest of the classifications associated with the other binary classifiers.

10. The system according to claim 8, where the processing performed by post-processing module (1026) comprises at least one of:

a) determining cardinality for the plurality of classification results;

b) determining entropy for the plurality of classification results;

c) determining the difference between the highest and second highest values of the classification results; and

d) determining the separation between the rank orders of the highest and second highest values of the classification results.

11. A process for underwriting an insurance application based on a plurality of previous insurance application underwriting decisions, where underwriting the insurance application includes assigning a classification to the insurance application, the process comprising:

receiving the insurance application, where the insurance application has a plurality of features;

processing a plurality of features associated with the insurance application;

generating a processed result for each of the plurality of features;

generating a classification result for the insurance application based on the processed result;

processing the classification results;

generating a processed classification result; and

selecting one of the processed classification results, where the selection indicates a classification assignment for the insurance application and selects the highest of the processed classification results.

12. The process according to claim 11, where the classification results are generated by a plurality of binary classifiers (5010, 5020, 5030), and where there is a classifier for each classification.

13. The process according to claim 12, where the classification results from each of the plurality of binary classifiers (5010, 5020, 5030) comprise one of:

a) the selection of the classification associated with the binary classifier; and

b) a selection of the rest of the classifications associated with the other binary classifiers.

14. The process according to claim 11, where processing a plurality of features further comprises at least one of:

a) range normalization; and

b) feature extraction.

15. The process according to claim 11, where processing the classification results further comprises generating a tag, wherein the tag is generated based on one or more rules applied to at least one of the plurality of features.

16. The process according to claim 11, where processing the classification results further comprises at least one of:

a) determining cardinality for the plurality of classification results;

b) determining entropy for the plurality of classification results;

c) determining the difference between the highest and second highest values of the classification results; and

d) determining the separation between the rank orders of the highest and second highest values of the classification results.

17. The process according to claim 11, where selecting one of the processed classification results further comprises a maximum classification selector (5040) which selects the highest of the processed classification results.

18. A process for underwriting an insurance application based on a plurality of previous insurance application underwriting decisions, where underwriting the insurance application includes assigning a classification to the insurance application, the process comprising:

receiving the insurance application, where the insurance application has a plurality of features;

processing a plurality of features associated with the insurance application, where processing a plurality of features further comprises at least one of:

a) range normalization; and

b) generating a tag, wherein the tag is generated based on one or more rules applied to at least one of the plurality of features;

generating a processed result for each of the plurality of features;

generating a classification result for the insurance application based on the processed result, where the classification results are generated by a plurality of binary classifiers (5010, 5020, 5030), and where there is a classifier for each classification;

processing the classification results;

generating a processed classification result; and

selecting one of the processed classification results, where the selection indicates a classification assignment for the insurance application and selects the highest of the processed classification results.

19. The process according to claim 18, where the classification results from each of the plurality of binary classifiers (5010, 5020, 5030) comprise one of:

a) the selection of the classification associated with the binary classifier; and

b) a selection of the rest of the classifications associated with the other binary classifiers.

20. The process according to claim 18, where processing the classification results further comprises at least one of:

a) determining cardinality for the plurality of classification results;

b) determining entropy for the plurality of classification results;

c) determining the difference between the highest and second highest values of the classification results; and

d) determining the separation between the rank orders of the highest and second highest values of the classification results.

21. A system for underwriting an insurance application based on a plurality of previous insurance application underwriting decisions, where underwriting the

insurance application includes assigning a classification to the insurance application, the system comprising:

means for receiving the insurance application, where the insurance application has a plurality of features;

means for processing a plurality of features associated with the insurance application;

means for generating a processed result for each of the plurality of features;

means for generating a classification result (116) for the insurance application based on the processed result;

means for processing the classification results (126);

means for generating a processed classification result (126); and

means for selecting one of the processed classification results (5040), where the selection indicates a classification assignment for the insurance application and selects the highest of the processed classification results.

22. The system according to claim 21, where the classification results are generated by a plurality of binary classifiers (5010, 5020, 5030), and where there is a classifier for each classification.

23. The system according to claim 22, where the classification results from each of the plurality of binary classifiers (5010, 5020, 5030) comprise one of:

a) the selection of the classification associated with the binary classifier; and

b) a selection of the rest of the classifications associated with the other binary classifiers.

24. The system according to claim 21, where the means for processing a plurality of features further comprises at least one of:

a) range normalization; and

b) feature extraction.

25. The system according to claim 21, where the processing by the means for processing the classification results (126) further comprises means for generating a tag, wherein the tag is generated based on one or more rules applied to at least one of the plurality of features.

26. The system according to claim 21, where the means for processing the classification results (126) further comprises at least one of:

a) means for determining cardinality for the plurality of classification results;

b) means for determining entropy for the plurality of classification results;

c) means for determining the difference between the highest and second highest values of the classification results; and

d) means for determining the separation between the rank orders of the highest and second highest values of the classification results.

27. The system according to claim 21, where the means for selecting is a maximum classification selector (5040) which selects the highest of the processed classification results.

28. A system for underwriting an insurance application based on a plurality of previous insurance application underwriting decisions, where underwriting the insurance application includes assigning a classification to the insurance application, the system comprising:

means for receiving the insurance application, where the insurance application has a plurality of features;

means for processing a plurality of features associated with the insurance application, where processing a plurality of features further comprises at least one of:

a) range normalization; and

b) generating a tag, wherein the tag is generated based on one or more rules applied to at least one of the plurality of features;

means for generating a processed result for each of the plurality of features;

means for generating a classification result (116) for the insurance application based on the processed result, where the classification results are generated by a plurality of binary classifiers (5010, 5020, 5030), and where there is a classifier for each classification;

means for processing the classification results (126);

means for generating a processed classification result (126); and

means for selecting one of the processed classification results (5040), where the selection indicates a classification assignment for the insurance application and selects the highest of the processed classification results.

29. The system according to claim 28, where the classification results from each of the plurality of binary classifiers (5010, 5020, 5030) comprise one of:

a) the selection of the classification associated with the binary classifier; and

b) a selection of the rest of the classifications associated with the other binary classifiers.

30. The system according to claim 28, where the means for processing the classification results (126) further comprises at least one of:

a) means for determining cardinality for the plurality of classification results;

b) means for determining entropy for the plurality of classification results;

c) means for determining the difference between the highest and second highest values of the classification results; and

d) means for determining the separation between the rank orders of the highest and second highest values of the classification results.

31. A computer readable medium having code for causing a processor to underwrite an insurance application based on a plurality of previous insurance application underwriting decisions, where underwriting the insurance application includes assigning a classification to the insurance application, the computer medium comprising:

code for receiving the insurance application, where the insurance application has a plurality of features;

code for processing a plurality of features associated with the insurance application;

code for generating a processed result for each of the plurality of features;

code for generating a classification result (116) for the insurance application based on the processed result;

code for processing the classification results (126);

code for generating a processed classification result (126); and

code for selecting one of the processed classification results (5040), where the selection indicates a classification assignment for the insurance application and selects the highest of the processed classification results.

32. The medium according to claim 31, where the classification results are generated by a plurality of binary classifiers (5010, 5020, 5030), and where there is a classifier for each classification.

33. The medium according to claim 32, where the classification results from each of the plurality of binary classifiers (5010, 5020, 5030) comprise one of:

a) the selection of the classification associated with the binary classifier; and

b) a selection of the rest of the classifications associated with the other binary classifiers.

34. The medium according to claim 31, where the code for processing a plurality of features further comprises at least one of:

a) range normalization; and

b) feature extraction.

35. The medium according to claim 31, where the code for processing the classification results (126) further comprises code for generating a tag, wherein the tag is generated based on one or more rules applied to at least one of the plurality of features.

36. The medium according to claim 31, where the code for processing the classification results (126) further comprises at least one of:

a) code for determining cardinality for the plurality of classification results;

b) code for determining entropy for the plurality of classification results;

c) code for determining the difference between the highest and second highest values of the classification results; and

d) code for determining the separation between the rank orders of the highest and second highest values of the classification results.

37. The medium according to claim 31, where the code for selecting is a maximum classification selector (5040) which selects the highest of the processed classification results.

38. A computer medium having code for causing a processor to underwrite an insurance application based on a plurality of previous insurance application underwriting decisions, where underwriting the insurance application includes assigning a classification to the insurance application, the medium comprising:

code for receiving the insurance application, where the insurance application has a plurality of features;

code for processing a plurality of features associated with the insurance application, where processing a plurality of features further comprises at least one of:

a) range normalization; and

b) generating a tag, wherein the tag is generated based on one or more rules applied to at least one of the plurality of features;

code for generating a processed result for each of the plurality of features;

code for generating a classification result (116) for the insurance application based on the processed result, where the classification results are generated by a plurality of binary classifiers (5010, 5020, 5030), and where there is a classifier for each classification;

code for processing the classification results (126);

code for generating a processed classification result (126); and code for selecting one of the processed classification results (5040), where the selection indicates a classification assignment for the insurance application and selects the highest of the processed classification results.

39. The medium according to claim 38, where the classification results from each of the plurality of binary classifiers (5010, 5020, 5030) comprise one of:

a) the selection of the classification associated with the binary classifier; and

b) a selection of the rest of the classifications associated with the other binary classifiers.

40. The medium according to claim 38, where the code for processing the classification results (126) further comprises at least one of:

a) means for determining cardinality for the plurality of classification results;

b) means for determining entropy for the plurality of classification results;

c) means for determining the difference between the highest and second highest values of the classification results; and

d) means for determining the separation between the rank orders of the highest and second highest values of the classification results.

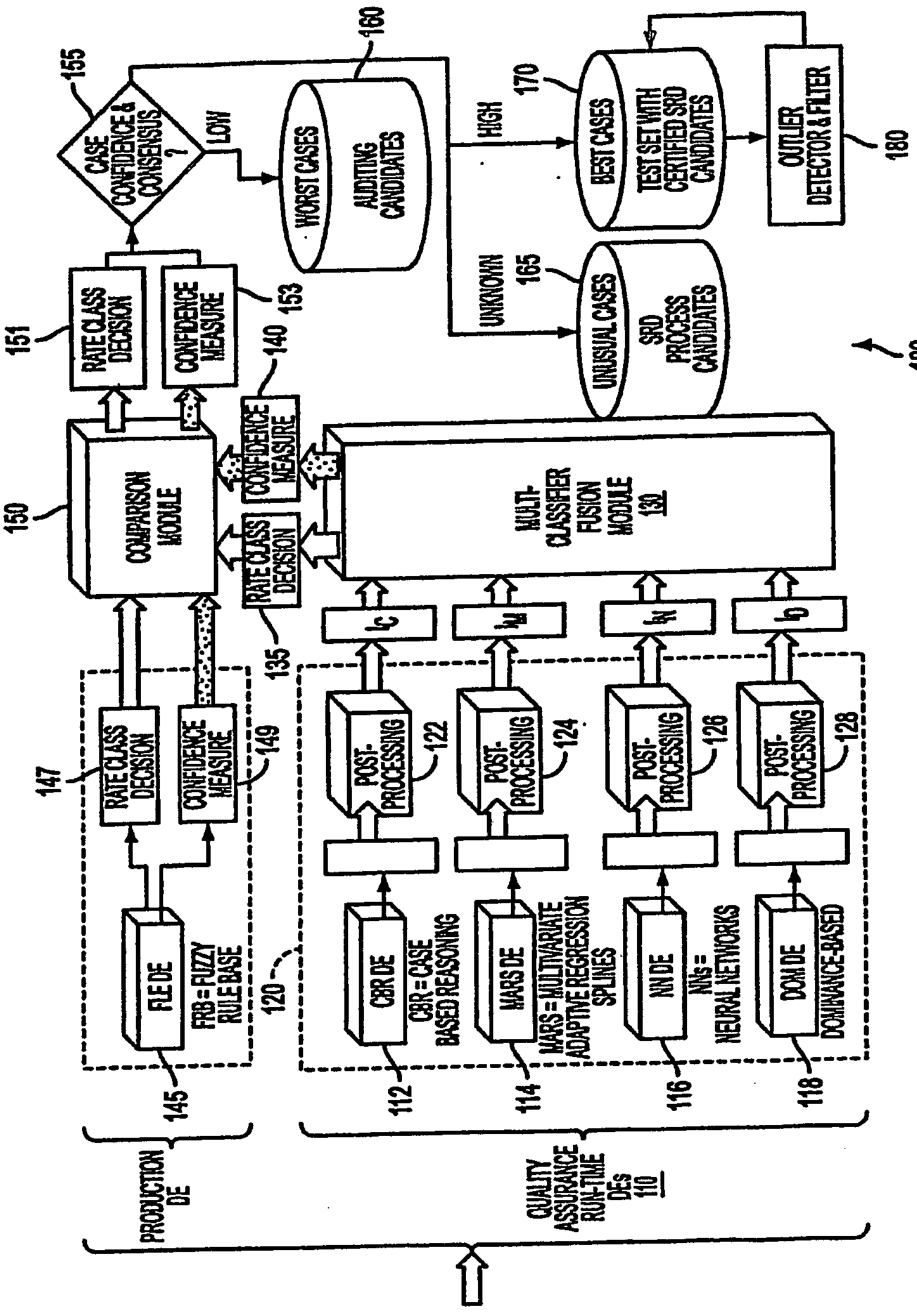

PRODUCTION DE
QUALITY ASSURANCE RUN-TIME DEs 110
145
FLE DE
FRB = FUZZY RULE BASE
147
RATE CLASS DECISION
CONFIDENCE MEASURE
149
120
CBR DE
CBR = CASE BASED REASONING
112
MARS DE
MARS = MULTIVARIATE ADAPTIVE REGRESSION SPLINES
114
NN DE
NNs = NEURAL NETWORKS
116
DOM DE
DOMINANCE-BASED
118
POST-PROCESSING
122
POST-PROCESSING
124
POST-PROCESSING
126
POST-PROCESSING
128
Ic
IM
IN
ID
MULTI-CLASSIFIER FUSION MODULE 130
RATE CLASS DECISION
135
CONFIDENCE MEASURE
140
COMPARISON MODULE
150
RATE CLASS DECISION
151
CONFIDENCE MEASURE
153
CASE CONFIDENCE & CONSENSUS ?
155
LOW
WORST CASES
AUDITING CANDIDATES
160
HIGH
BEST CASES
TEST SET WITH CERTIFIED SRD CANDIDATES
170
UNKNOWN
165
UNUSUAL CASES
SRD PROCESS CANDIDATES
OUTLIER DETECTOR & FILTER
180
100

FIG. 1

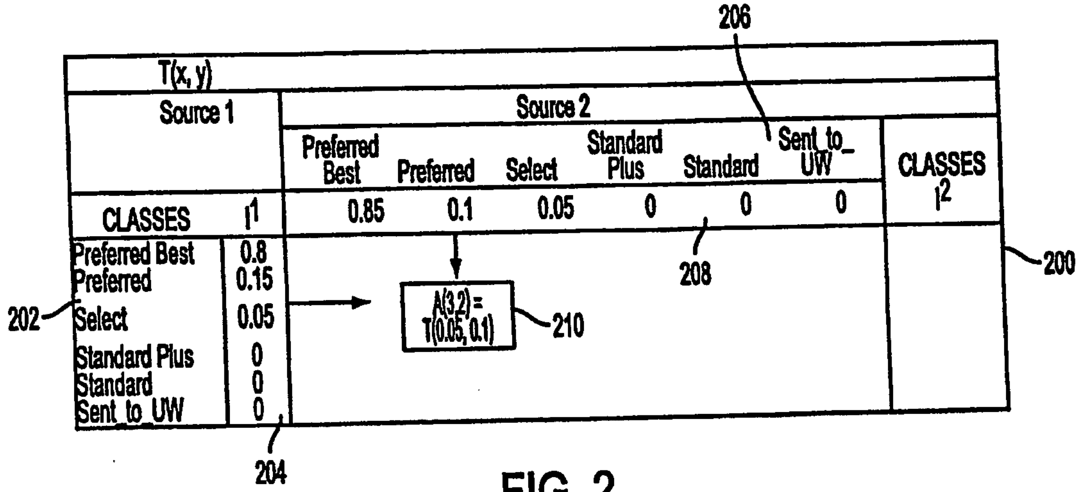

T(x, y)
Source 1
Source 2
Preferred Best
Preferred
Select
Standard Plus
Standard
Sent_to_UW
CLASSES l^2
CLASSES l^1
0.85
0.1
0.05
0
0
0
Preferred Best
Preferred
Select
Standard Plus
Standard
Sent_to_UW
0.8
0.15
0.05
0
0
0
A(3,2) = T(0.05, 0.1)
200
202
204
206
208
210

FIG. 2

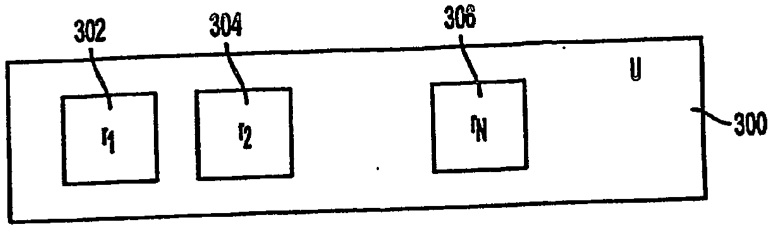
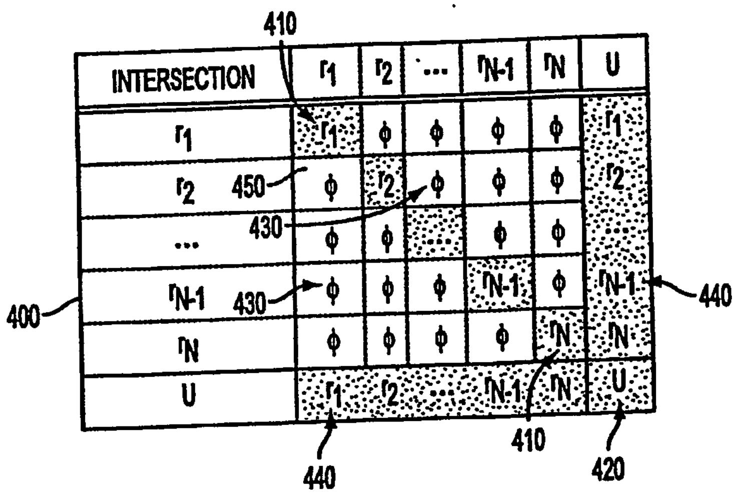


FIG. 3


410
450
430
430
400
440
440
410
420

INTERSECTION	r_1	r_2	...	r_{N-1}	r_N	U
r_1	r_1	φ	φ	φ	φ	r_1
r_2	φ	r_2	φ	φ	φ	r_2
...	φ	φ		φ	φ	
r_{N-1}	φ	φ	φ	r_{N-1}	φ	r_{N-1}
r_N	φ	φ	φ	φ	r_N	r_N
U	r_1	r_2	...	r_{N-1}	r_N	U

FIG. 4

T3 = min(x,y)		D(M2)					
D(M1)		Preferred Best	Preferred	Select	Standard Plus	Standard	Sent_to_UW
		0.9	0.05	0.05	0	0	0
Preferred Best	0.8	0.800	0.050	0.050	0	0	0.000
Preferred	0.15	0.150	0.050	0.050	0	0	0.000
Select	0.05	0.050	0.050	0.050	0	0	0.000
StandardPlus	0	0	0	0	0	0	0
Standard	0	0	0	0	0	0	0
Sent_to_UW	0	0	0	0	0	0	0

FIG. 5

T2.5 = [x^(-1)+y^(-1)-1]^(-1)		D(M2)					
D(M1)		Preferred Best	Preferred	Select	Standard Plus	Standard	Sent_to_UW
		0.9	0.05	0.05	0	0	0
Preferred Best	0.8	0.735	0.049	0.049	0	0	0.000
Preferred	0.15	0.148	0.039	0.039	0	0	0.000
Select	0.05	0.050	0.026	0.026	0	0	0.000
StandardPlus	0	0	0	0	0	0	0
Standard	0	0	0	0	0	0	0
Sent_to_UW	0	0	0	0	0	0	0

FIG. 6

T2 = x * y		D(M2)					
D(M1)		Preferred Best	Preferred	Select	Standard Plus	Standard	Sent_to_UW
		0.9	0.05	0.05	0	0	0
Preferred Best	0.8	0.720	0.040	0.040	0	0	0.000
Preferred	0.15	0.135	0.008	0.008	0	0	0.000
Select	0.05	0.045	0.003	0.003	0	0	0.000
StandardPlus	0	0	0	0	0	0	0
Standard	0	0	0	0	0	0	0
Sent_to_UW	0	0	0	0	0	0	0

FIG. 7

T3 = min(x,y)		D(M2)					
D(M1)		Preferred Best	Preferred	Select	StandardPlus	Standard	Sent_to_UW
		0.9	0.05	0.05	0	0	0
Preferred Best	0.8	0.615	0.038	0.038	0	0	0.000
Preferred	0.15	0.115	0.038	0.038	0	0	0.000
Select	0.05	0.038	0.038	0.038	0	0	0.000
StandardPlus	0	0	0	0	0	0	0
Standard	0	0	0	0	0	0	0
Sent_to_UW	0	0	0	0	0	0	0

FIG. 10

T2.5 = [x^(-1)+y^(-1)-1]^(-1)		D(M2)					
D(M1)		Preferred Best	Preferred	Select	StandardPlus	Standard	Sent_to_UW
		0.9	0.05	0.05	0	0	0
Preferred Best	0.8	0.633	0.043	0.043	0	0	0.000
Preferred	0.15	0.127	0.034	0.034	0	0	0.000
Select	0.05	0.043	0.022	0.022	0	0	0.000
StandardPlus	0	0	0	0	0	0	0
Standard	0	0	0	0	0	0	0
Sent_to_UW	0	0	0	0	0	0	0

FIG. 11

T2 = x * y		D(M2)					No Decision:
D(M1)		Preferred Best	Preferred	Select	StandardPlus	Standard	Sent_to_UW
		0.9	0.05	0.05	0	0	0
Preferred Best	0.8	0.720	0.040	0.040	0	0	0.000
Preferred	0.15	0.135	0.008	0.008	0	0	0.000
Select	0.05	0.045	0.003	0.003	0	0	0.000
StandardPlus	0	0	0	0	0	0	0
Standard	0	0	0	0	0	0	0
Sent_to_UW	0	0	0	0	0	0	0

FIG. 12

T1.5 = max(0, (x^0.5+y^0.5-1))^2		D(M2)					
D(M1)		Preferred Best	Preferred	Select	Standard Plus	Standard	Sent_to_UW
		0.9	0.05	0.05	0	0	0
Preferred Best	0.8	0.711	0.014	0.014	0	0	0.000
Preferred	0.15	0.113	0.000	0.000	0	0	0.000
Select	0.05	0.030	0.000	0.000	0	0	0.000
StandardPlus	0	0	0	0	0	0	0
Standard	0	0	0	0	0	0	0
Sent_to_UW	0	0	0	0	0	0	0

FIG. 8

T1.5 = max(0, (x^0.5+y^0.5-1))^2		D(M2)					No Decision:
D(M1)		Preferred Best	Preferred	Select	StandardPlus	Standard	Sent_to_UW
		0.9	0.05	0.05	0	0	0
Preferred Best	0.8	0.807	0.016	0.016	0	0	0.000
Preferred	0.15	0.128	0.000	0.000	0	0	0.000
Select	0.05	0.034	0.000	0.000	0	0	0.000
StandardPlus	0	0	0	0	0	0	0
Standard	0	0	0	0	0	0	0
Sent_to_UW	0	0	0	0	0	0	0

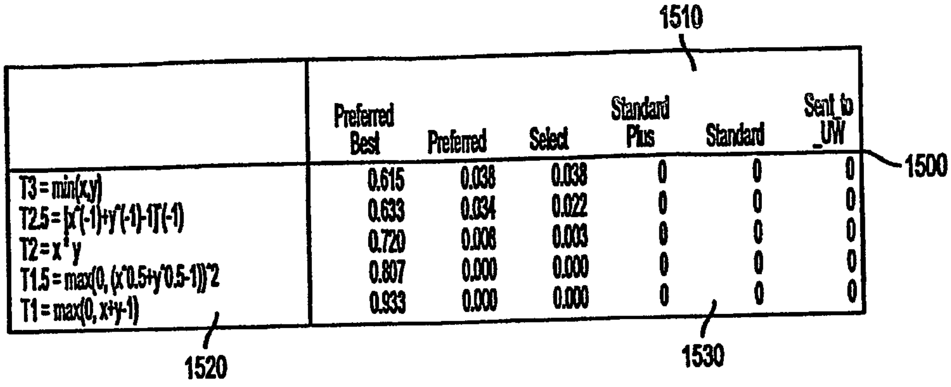
FIG. 13

T1 = max(0, x+y-1)		D(M2)					
D(M1)		Preferred Best	Preferred	Select	Standard Plus	Standard	Sent_to_UW
		0.9	0.05	0.05	0	0	0
Preferred Best	0.8	0.700	0.000	0.000	0	0	0.000
Preferred	0.15	0.050	0.000	0.000	0	0	0.000
Select	0.05	0.000	0.000	0.000	0	0	0.000
StandardPlus	0	0	0	0	0	0	0
Standard	0	0	0	0	0	0	0
Sent_to_UW	0	0	0	0	0	0	0

FIG. 9

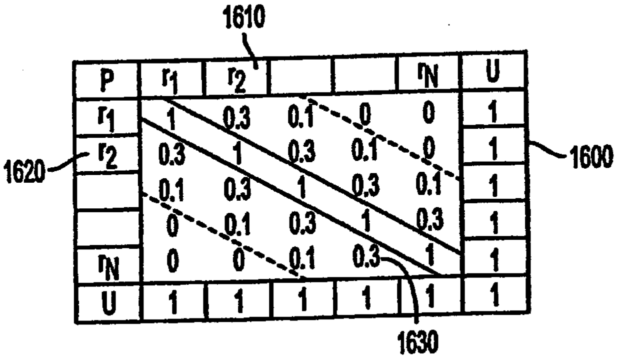
T1 = max(0, x+y-1)		D(M2)					No Decision:
D(M1)		Preferred Best	Preferred	Select	StandardPlus	Standard	Sent_to_UW
		0.9	0.05	0.05	0	0	0
Preferred Best	0.8	0.933	0.000	0.000	0	0	0.000
Preferred	0.15	0.067	0.000	0.000	0	0	0.000
Select	0.05	0.000	0.000	0.000	0	0	0.000
StandardPlus	0	0	0	0	0	0	0
Standard	0	0	0	0	0	0	0
Sent_to_UW	0	0	0	0	0	0	0

FIG. 14


1510
1500
1520
1530

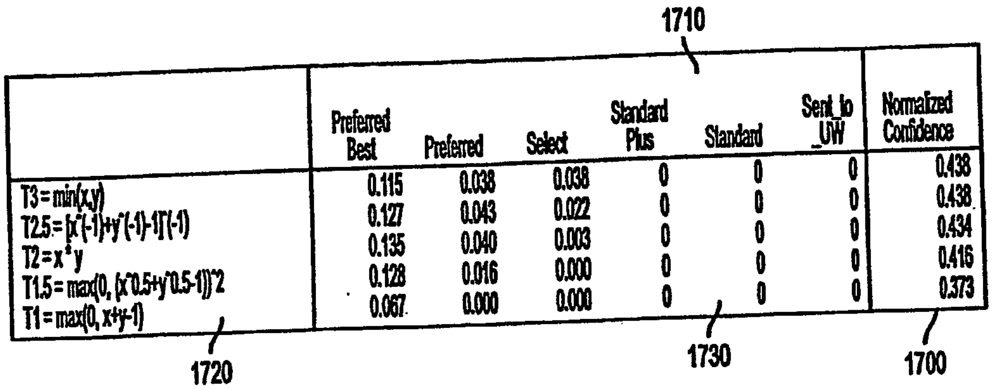
	Preferred Best	Preferred	Select	Standard Plus	Standard	Sent to _UW
T3 = min(x,y)	0.615	0.038	0.038	0	0	0
T2.5 = [x^(-1)+y^(-1)-1]^(-1)	0.633	0.034	0.022	0	0	0
T2 = x * y	0.720	0.008	0.003	0	0	0
T1.5 = max(0, (x^0.5+y^0.5-1))^2	0.807	0.000	0.000	0	0	0
T1 = max(0, x+y-1)	0.933	0.000	0.000	0	0	0

FIG. 15


1610
1620
1600
1630

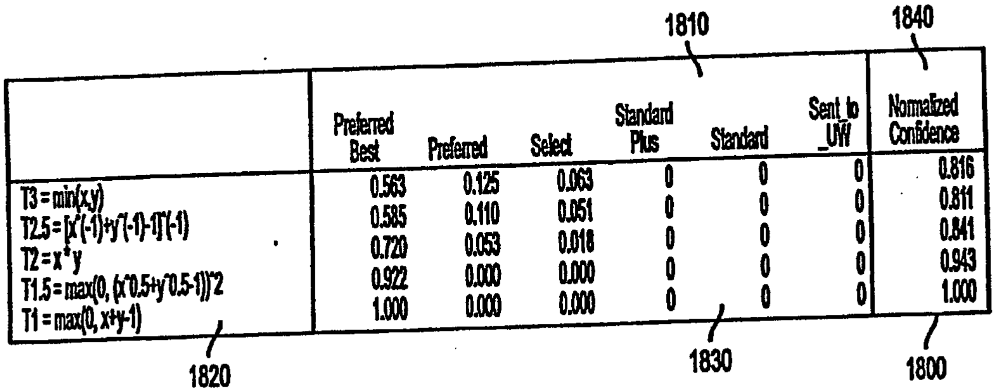
P	r_1	r_2			r_N	U
r_1	1	0.3	0.1	0	0	1
r_2	0.3	1	0.3	0.1	0	1
	0.1	0.3	1	0.3	0.1	1
	0	0.1	0.3	1	0.3	1
r_N	0	0	0.1	0.3	1	1
U	1	1	1	1	1	1

FIG. 16


1710
1720
1730
1700

	Preferred Best	Preferred	Select	Standard Plus	Standard	Sent_to _UW	Normalized Confidence
T3 = min(x,y)	0.115	0.038	0.038	0	0	0	0.438
T2.5 = [x^(-1)+y^(-1)-1]^(-1)	0.127	0.043	0.022	0	0	0	0.438
T2 = x * y	0.135	0.040	0.003	0	0	0	0.434
T1.5 = max(0, (x^0.5+y^0.5-1))^2	0.128	0.016	0.000	0	0	0	0.416
T1 = max(0, x+y-1)	0.067	0.000	0.000	0	0	0	0.373

FIG. 17


1810
1840
1820
1830
1800

	Preferred Best	Preferred	Select	Standard Plus	Standard	Sent_to _UW	Normalized Confidence
T3 = min(x,y)	0.563	0.125	0.063	0	0	0	0.816
T2.5 = [x^(-1)+y^(-1)-1]^(-1)	0.585	0.110	0.051	0	0	0	0.811
T2 = x * y	0.720	0.053	0.018	0	0	0	0.841
T1.5 = max(0, (x^0.5+y^0.5-1))^2	0.922	0.000	0.000	0	0	0	0.943
T1 = max(0, x+y-1)	1.000	0.000	0.000	0	0	0	1.000

FIG. 18

T3 = min(x,y)		D(M2)					
D(M1)		Preferred Best	Preferred	Select	Standard Plus	Standard	Sent_to_UW
		0.6	0.05	0.05	0	0	0.3
Preferred Best	0.8	0.600	0.050	0.050	0	0	0.300
Preferred	0.15	0.150	0.050	0.050	0	0	0.150
Select	0.05	0.050	0.050	0.050	0	0	0.050
StandardPlus	0	0	0	0	0	0	0
Standard	0	0	0	0	0	0	0
Sent_to_UW	0	0	0	0	0	0	0

FIG. 19

$T2.5 = [x^{(-1)}+y^{(-1)}-1]^{(-1)}$		D(M2)					
D(M1)		Preferred Best	Preferred	Select	Standard Plus	Standard	Sent_to_UW
		0.6	0.05	0.05	0	0	0.3
Preferred Best	0.8	0.522	0.049	0.049	0	0	0.279
Preferred	0.15	0.136	0.039	0.039	0	0	0.111
Select	0.05	0.048	0.026	0.026	0	0	0.045
StandardPlus	0	0	0	0	0	0	0
Standard	0	0	0	0	0	0	0
Sent_to_UW	0	0	0	0	0	0	0

FIG. 20

T2 = x*y		D(M2)					
D(M1)		Preferred Best	Preferred	Select	Standard Plus	Standard	Sent_to_UW
		0.6	0.05	0.05	0	0	0.3
Preferred Best	0.8	0.480	0.040	0.040	0	0	0.240
Preferred	0.15	0.090	0.008	0.008	0	0	0.045
Select	0.05	0.030	0.003	0.003	0	0	0.015
StandardPlus	0	0	0	0	0	0	0
Standard	0	0	0	0	0	0	0
Sent_to_UW	0	0	0	0	0	0	0

FIG. 21

T3 = min(x,y)		D(M2)					
D(M1)		Preferred Best	Preferred	Select	StandardPlus	Standard	Sent_to_UW
		0.6	0.05	0.05	0	0	0.3
Preferred Best	0.8	0.375	0.031	0.031	0	0	0.188
Preferred	0.15	0.094	0.031	0.031	0	0	0.094
Select	0.05	0.031	0.031	0.031	0	0	0.031
StandardPlus	0	0	0	0	0	0	0
Standard	0	0	0	0	0	0	0
Sent_to_UW	0	0	0	0	0	0	0

FIG. 24

$T2.5 = [x^{(-1)}+y^{(-1)}-1]^{(-1)}$		D(M2)					
D(M1)		Preferred Best	Preferred	Select	StandardPlus	Standard	Sent_to_UW
		0.6	0.05	0.05	0	0	0.3
Preferred Best	0.8	0.381	0.036	0.036	0	0	0.204
Preferred	0.15	0.100	0.028	0.028	0	0	0.081
Select	0.05	0.035	0.019	0.019	0	0	0.033
StandardPlus	0	0	0	0	0	0	0
Standard	0	0	0	0	0	0	0
Sent_to_UW	0	0	0	0	0	0	0

FIG. 25

T2 = x*y		D(M2)					No Decision:
D(M1)		Preferred Best	Preferred	Select	StandardPlus	Standard	Sent_to_UW
		0.6	0.05	0.05	0	0	0.3
Preferred Best	0.8	0.480	0.040	0.040	0	0	0.240
Preferred	0.15	0.090	0.008	0.008	0	0	0.045
Select	0.05	0.030	0.003	0.003	0	0	0.015
StandardPlus	0	0	0	0	0	0	0
Standard	0	0	0	0	0	0	0
Sent_to_UW	0	0	0	0	0	0	0

FIG. 26

T1.5 = max(0, (x^0.5+y^0.5-1))^2		D(M2)					
D(M1)		Preferred Best	Preferred	Select	Standard Plus	Standard	Sent_to_UW
		0.6	0.05	0.05	0	0	0.3
Preferred Best	0.8	0.448	0.014	0.014	0	0	0.195
Preferred	0.15	0.026	0.000	0.000	0	0	0.000
Select	0.05	0.000	0.000	0.000	0	0	0.000
StandardPlus	0	0	0	0	0	0	0
Standard	0	0	0	0	0	0	0
Sent_to_UW	0	0	0	0	0	0	0

FIG. 22

T1 = max(0, x+y-1)		D(M2)					
D(M1)		Preferred Best	Preferred	Select	Standard Plus	Standard	Sent_to_UW
		0.6	0.05	0.05	0	0	0.3
Preferred Best	0.8	0.400	0.000	0.000	0	0	0.100
Preferred	0.15	0.000	0.000	0.000	0	0	0.000
Select	0.05	0.000	0.000	0.000	0	0	0.000
StandardPlus	0	0	0	0	0	0	0
Standard	0	0	0	0	0	0	0
Sent_to_UW	0	0	0	0	0	0	0

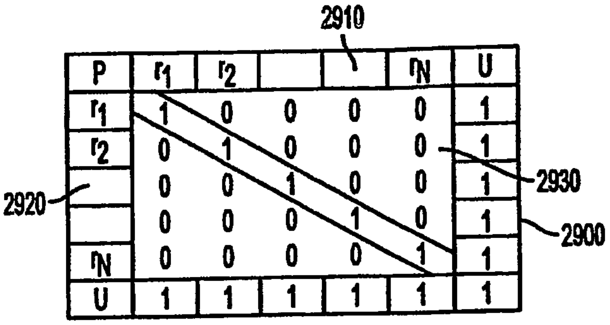
FIG. 23

T1.5 = max(0, (x^0.5+y^0.5-1))^2		D(M2)					No Decision:
D(M1)		Preferred Best	Preferred	Select	Standard Plus	Standard	Sent_to_UW
		0.6	0.05	0.05	0	0	0.3
Preferred Best	0.8	0.642	0.020	0.020	0	0	0.280
Preferred	0.15	0.038	0.000	0.000	0	0	0.000
Select	0.05	0.000	0.000	0.000	0	0	0.000
StandardPlus	0	0	0	0	0	0	0
Standard	0	0	0	0	0	0	0
Sent_to_UW	0	0	0	0	0	0	0

FIG. 27

T1 = max(0, x+y-1)		D(M2)					No Decision:
D(M1)		Preferred Best	Preferred	Select	Standard Plus	Standard	Sent_to_UW
		0.6	0.05	0.05	0	0	0.3
Preferred Best	0.8	0.800	0.000	0.000	0	0	0.2000
Preferred	0.15	0.000	0.000	0.000	0	0	0.0000
Select	0.05	0.000	0.000	0.000	0	0	0.0000
StandardPlus	0	0	0	0	0	0	0
Standard	0	0	0	0	0	0	0
Sent_to_UW	0	0	0	0	0	0	0

FIG. 28



P	r_1	r_2			r_N	U
r_1	1	0	0	0	0	1
r_2	0	1	0	0	0	1
	0	0	1	0	0	1
	0	0	0	1	0	1
r_N	0	0	0	0	1	1
U	1	1	1	1	1	1

FIG. 29

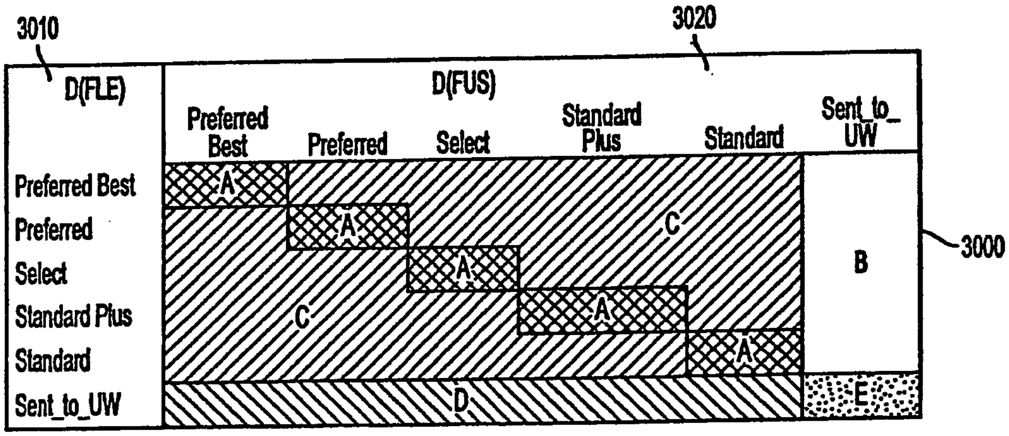


FIG. 30

3120

CONFIDENCE THRESHOLD T1	0	0.025	0.05	0.075	0.1	0.125	0.15
NO FUSION DECISIONS	36	80	97	122	148	182	212
AGREEMENTS	1669	1661	1652	1641	1623	1604	1588
FALSE POSITIVE	82	52	48	40	35	29	25
FALSE NEGATIVES	43	43	42	39	36	32	27
CORRECTIONS	37	34	32	29	29	26	22
COMPLETE DISAGREEMENT	8	5	4	4	4	2	1
TOTAL	1875	1875	1875	1875	1875	1875	1875

3100

3110 3130

FIG. 31

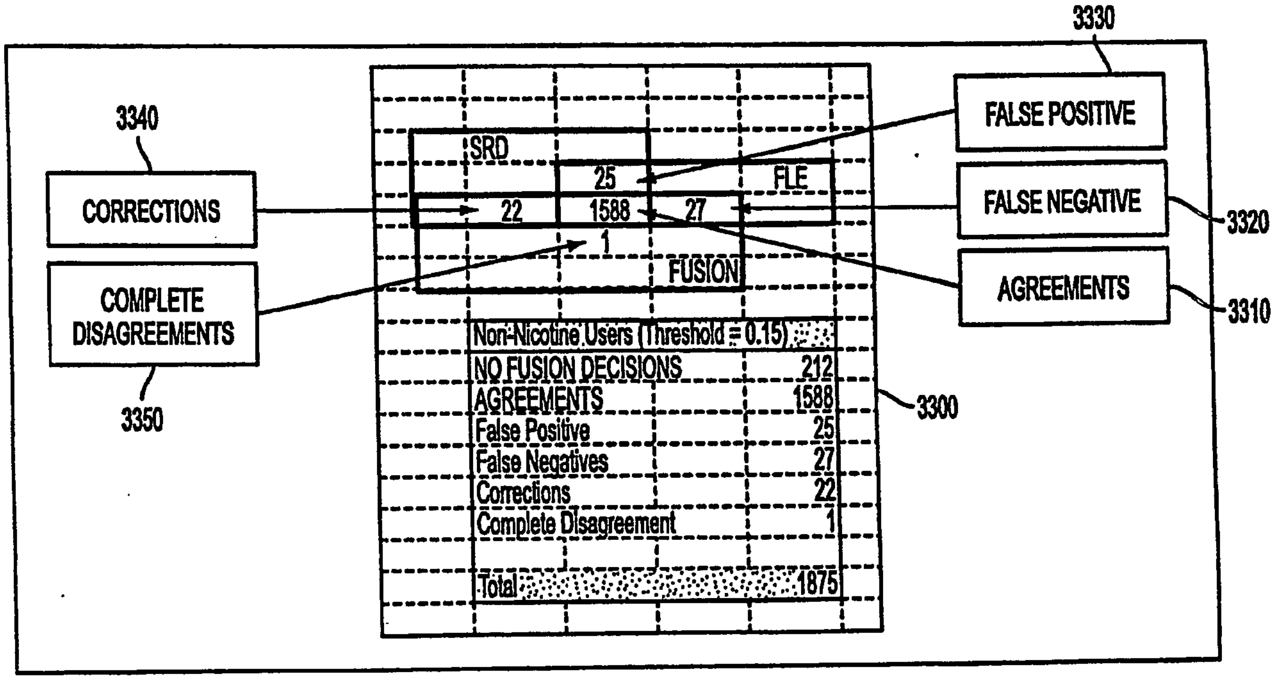
3220 3230

CONFIDENCE THRESHOLD	0	0.025	0.05	0.075	0.1	0.125	0.15	0.175	0.2	0.225	0.25
NO FUSION DECISIONS	37	47	55	56	59	62	68	75	84	90	94
AGREEMENTS	750	745	740	739	736	733	729	728	724	721	718
FALSE POSITIVE	23	20	18	18	18	18	16	13	11	10	9
FALSE NEGATIVES	16	15	15	15	15	15	15	13	11	9	9
CORRECTIONS	5	4	3	3	3	3	3	2	1	1	1
COMPLETE DISAGREEMENT	0	0	0	0	0	0	0	0	0	0	0
TOTAL	831	831	831	831	831	831	831	831	831	831	831

3210

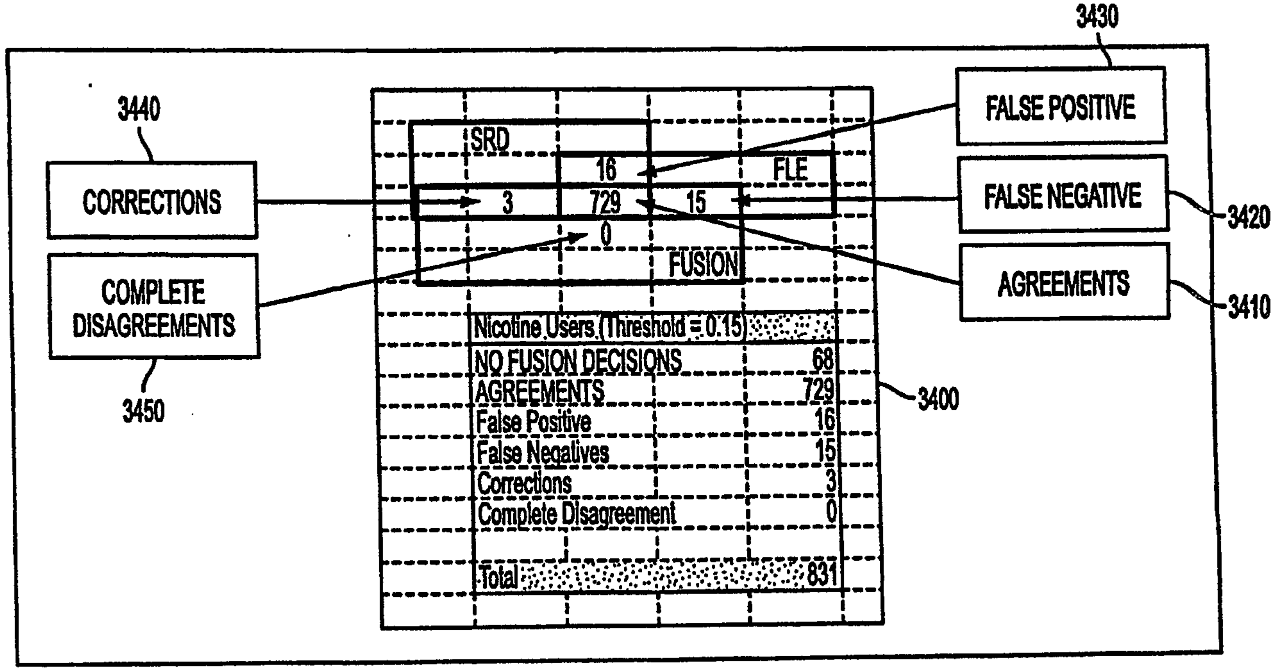
3200

FIG. 32



Non-Nicotine Users (Threshold = 0.15)	
NO FUSION DECISIONS	212
AGREEMENTS	1588
False Positive	25
False Negatives	27
Corrections	22
Complete Disagreement	1
Total	1875

FIG. 33



Nicotine Users (Threshold = 0.15)	
NO FUSION DECISIONS	68
AGREEMENTS	729
False Positive	16
False Negatives	15
Corrections	3
Complete Disagreement	0
Total	831

FIG. 34

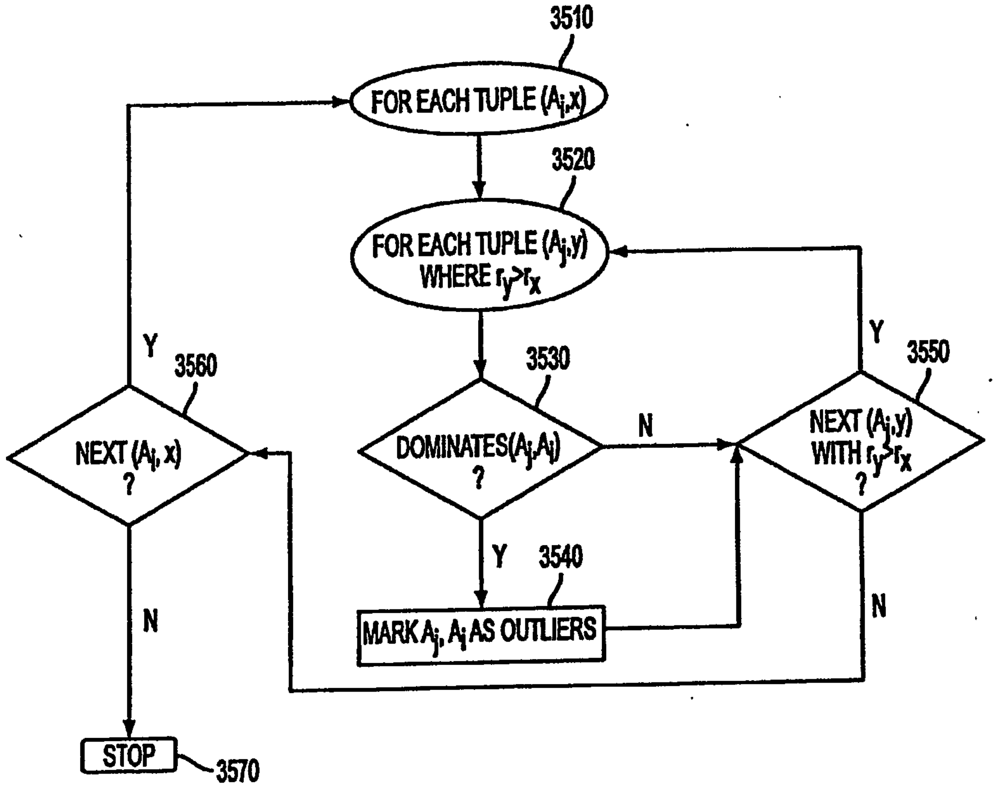

3510
FOR EACH TUPLE (A_i,x)
3520
FOR EACH TUPLE (A_j,y) WHERE r_y>r_x
3530
DOMINATES(A_j,A_i) ?
N
Y
3540
MARK A_j, A_i AS OUTLIERS
Y
3550
NEXT (A_j,y) WITH r_y>r_x ?
N
Y
3560
NEXT (A_i, x) ?
N
STOP
3570

FIG. 35

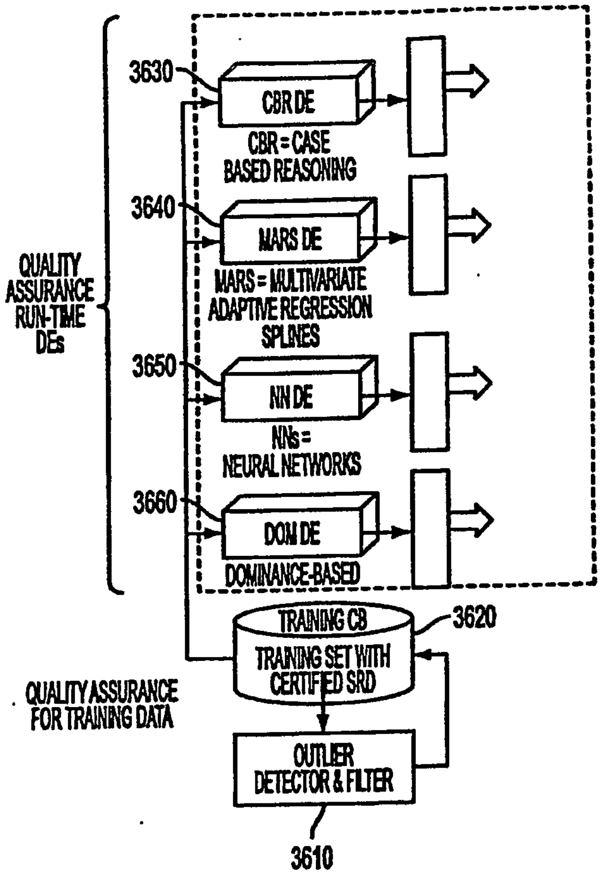

3630
CBR DE
CBR = CASE BASED REASONING
3640
MARS DE
MARS = MULTIVARIATE ADAPTIVE REGRESSION SPLINES
QUALITY ASSURANCE RUN-TIME DEs
3650
NN DE
NNs = NEURAL NETWORKS
3660
DOM DE
DOMINANCE-BASED
TRAINING CB
TRAINING SET WITH CERTIFIED SRD
3620
QUALITY ASSURANCE FOR TRAINING DATA
OUTLIER DETECTOR & FILTER
3610

FIG. 36

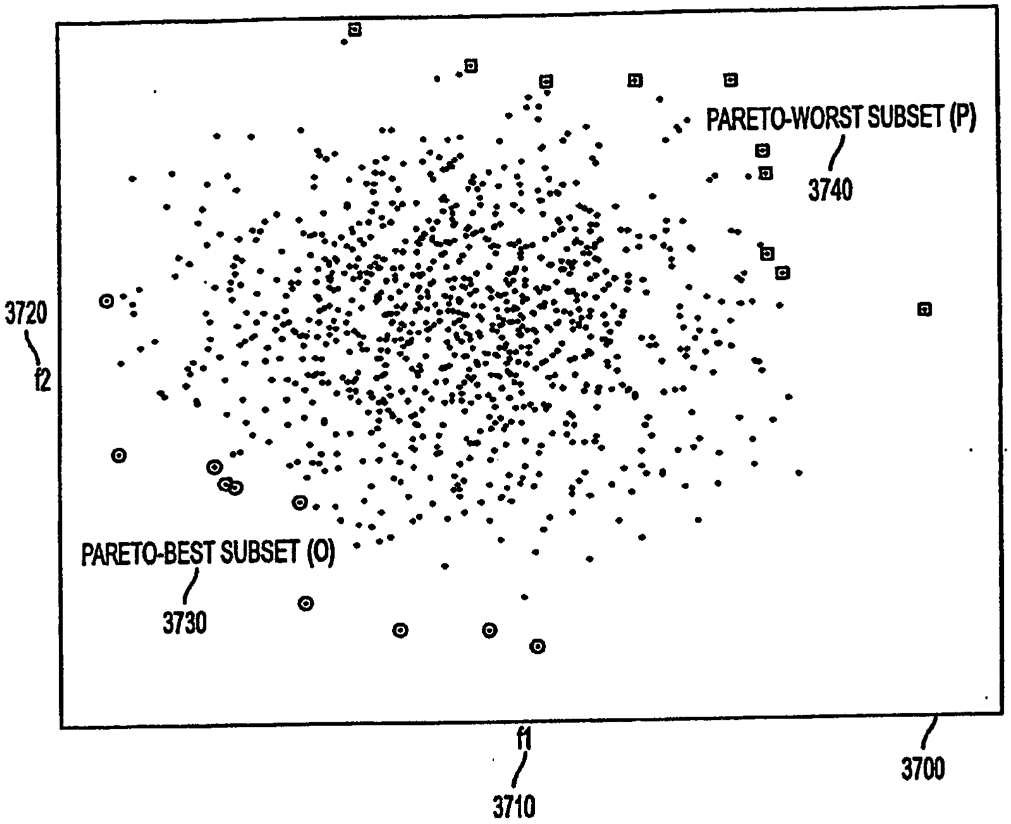

PARETO-WORST SUBSET (P)
3740
3720
f2
PARETO-BEST SUBSET (O)
3730
f1
3710
3700

FIG. 37

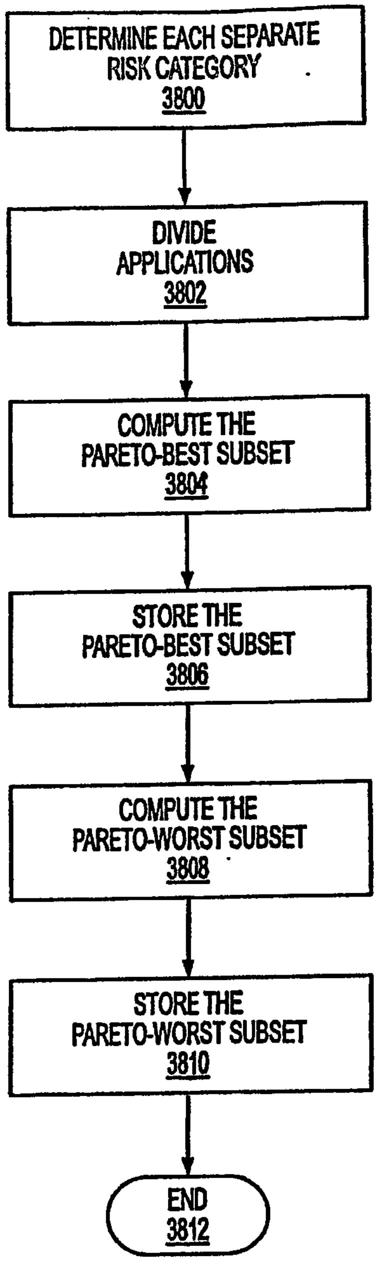

DETERMINE EACH SEPARATE RISK CATEGORY
3800
DIVIDE APPLICATIONS
3802
COMPUTE THE PARETO-BEST SUBSET
3804
STORE THE PARETO-BEST SUBSET
3806
COMPUTE THE PARETO-WORST SUBSET
3808
STORE THE PARETO-WORST SUBSET
3810
END
3812

FIG. 38

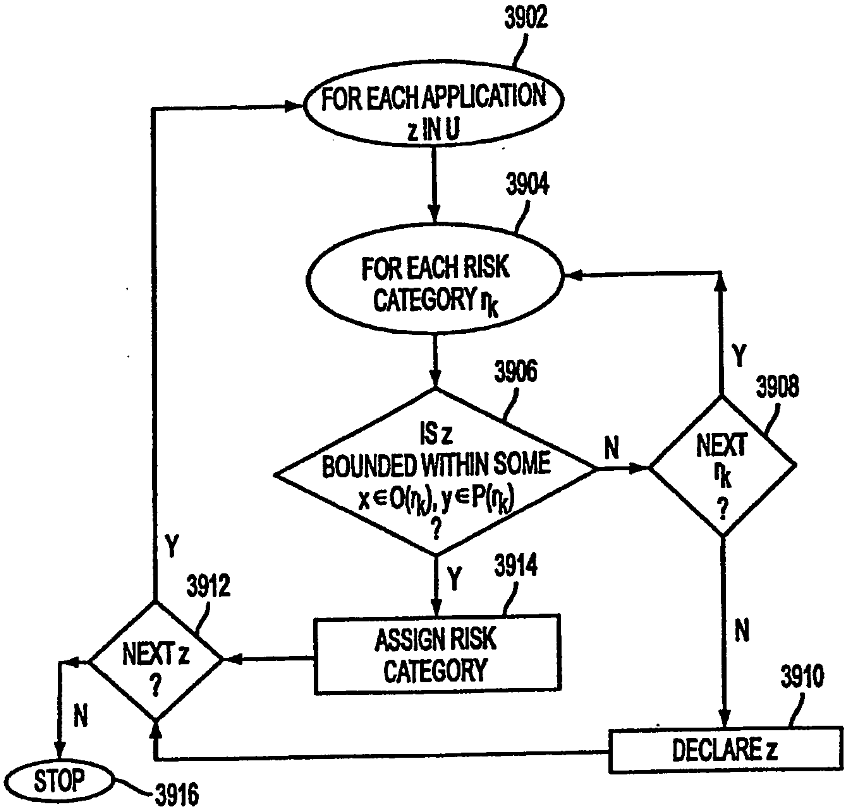
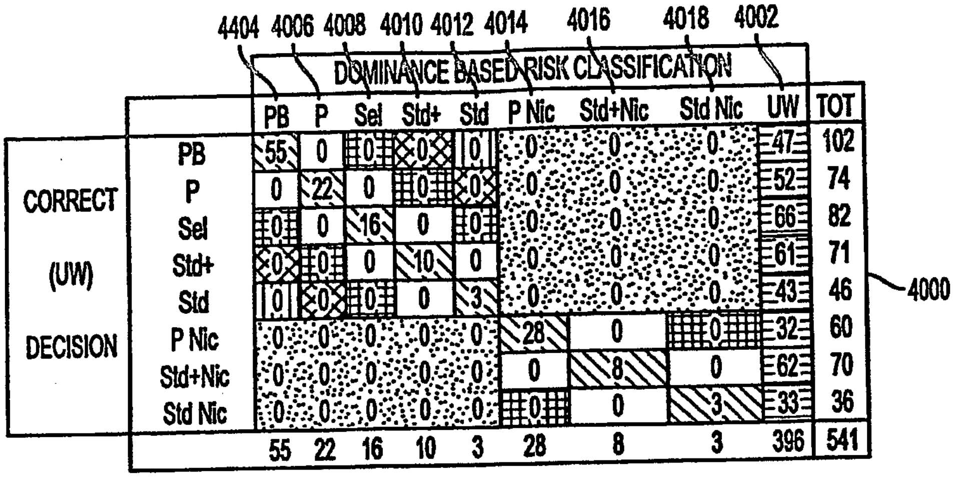
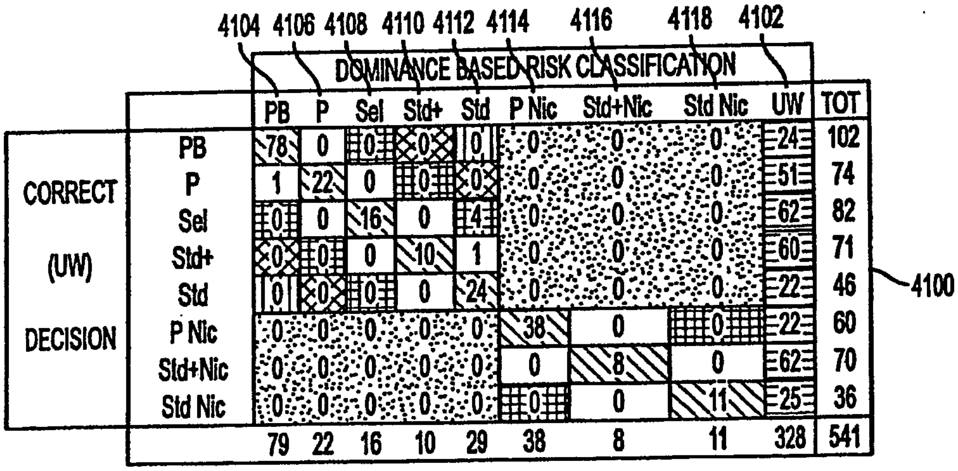

3902
FOR EACH APPLICATION z IN U
3904
FOR EACH RISK CATEGORY r_k
3906
IS z BOUNDED WITHIN SOME $x \in O(r_k)$, $y \in P(r_k)$?
N
Y
3908
NEXT r_k ?
Y
N
3910
DECLARE z
3914
ASSIGN RISK CATEGORY
Y
3912
NEXT z ?
Y
N
STOP
3916

FIG. 39


4404
4006
4008
4010
4012
4014
4016
4018
4002
4000

		DOMINANCE BASED RISK CLASSIFICATION									
		PB	P	Sel	Std+	Std	P Nic	Std+Nic	Std Nic	UW	TOT
CORRECT (UW) DECISION	PB	55	0	0	0	0	0	0	0	47	102
	P	0	22	0	0	0	0	0	0	52	74
	Sel	0	0	16	0	0	0	0	0	66	82
	Std+	0	0	0	10	0	0	0	0	61	71
	Std	0	0	0	0	3	0	0	0	43	46
	P Nic	0	0	0	0	0	28	0	0	32	60
	Std+Nic	0	0	0	0	0	0	8	0	62	70
	Std Nic	0	0	0	0	0	0	0	3	33	36
		55	22	16	10	3	28	8	3	396	541

FIG. 40


4104
4106
4108
4110
4112
4114
4116
4118
4102
4100

		DOMINANCE BASED RISK CLASSIFICATION									
		PB	P	Sel	Std+	Std	P Nic	Std+Nic	Std Nic	UW	TOT
CORRECT (UW) DECISION	PB	78	0	0	0	0	0	0	0	24	102
	P	1	22	0	0	0	0	0	0	51	74
	Sel	0	0	16	0	4	0	0	0	62	82
	Std+	0	0	0	10	1	0	0	0	60	71
	Std	0	0	0	0	24	0	0	0	22	46
	P Nic	0	0	0	0	0	38	0	0	22	60
	Std+Nic	0	0	0	0	0	0	8	0	62	70
	Std Nic	0	0	0	0	0	0	0	11	25	36
		79	22	16	10	29	38	8	11	328	541

FIG. 41

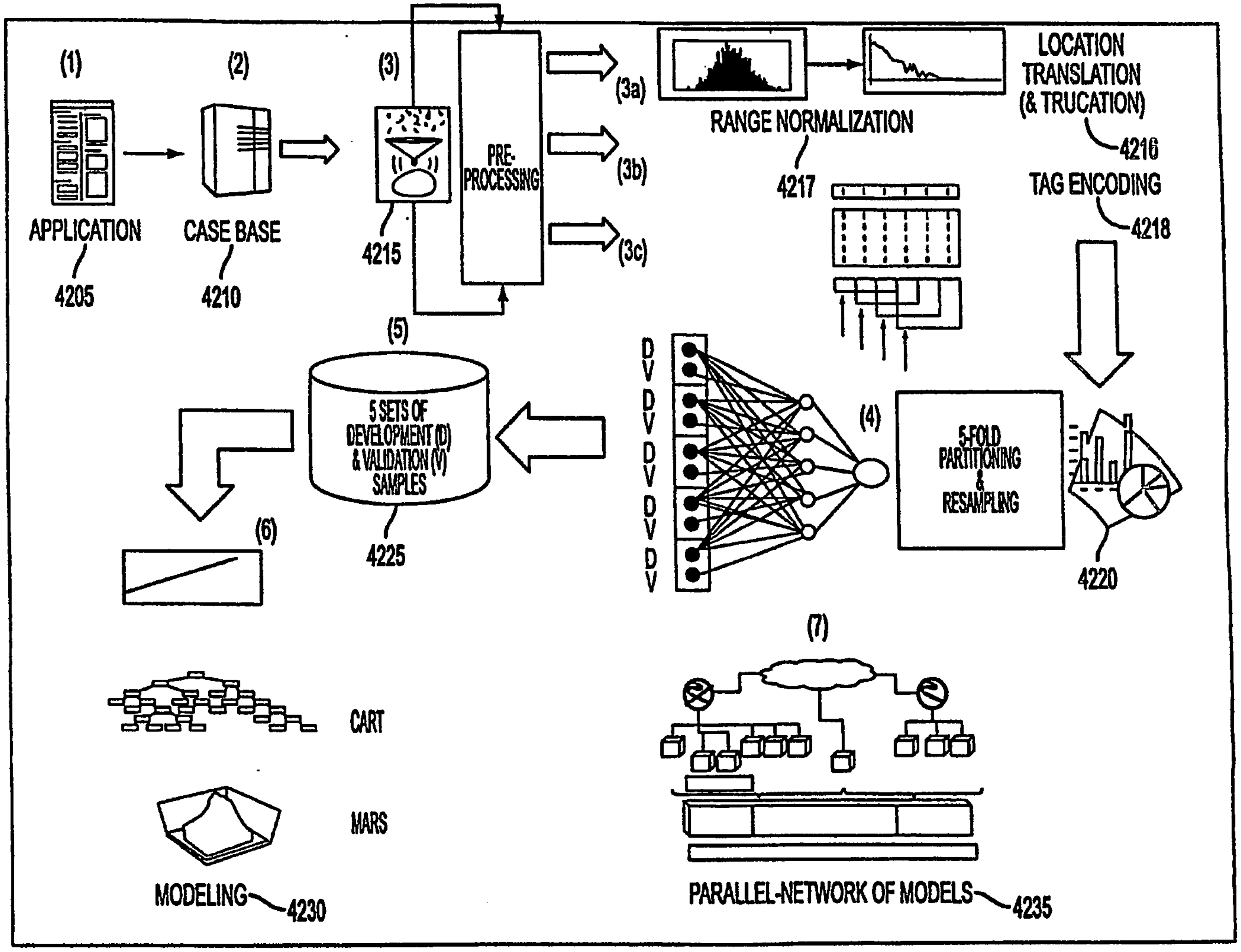

(1)
APPLICATION
4205
(2)
CASE BASE
4210
(3)
4215
PRE-PROCESSING
(3a)
(3b)
(3c)
RANGE NORMALIZATION
4217
LOCATION TRANSLATION (& TRUCATION)
4216
TAG ENCODING
4218
(4)
5-FOLD PARTITIONING & RESAMPLING
4220
D
V
D
V
D
V
D
V
D
V
(5)
5 SETS OF DEVELOPMENT (D) & VALIDATION (V) SAMPLES
4225
(6)
CART
MARS
MODELING
4230
(7)
PARALLEL-NETWORK OF MODELS
4235

FIG. 42

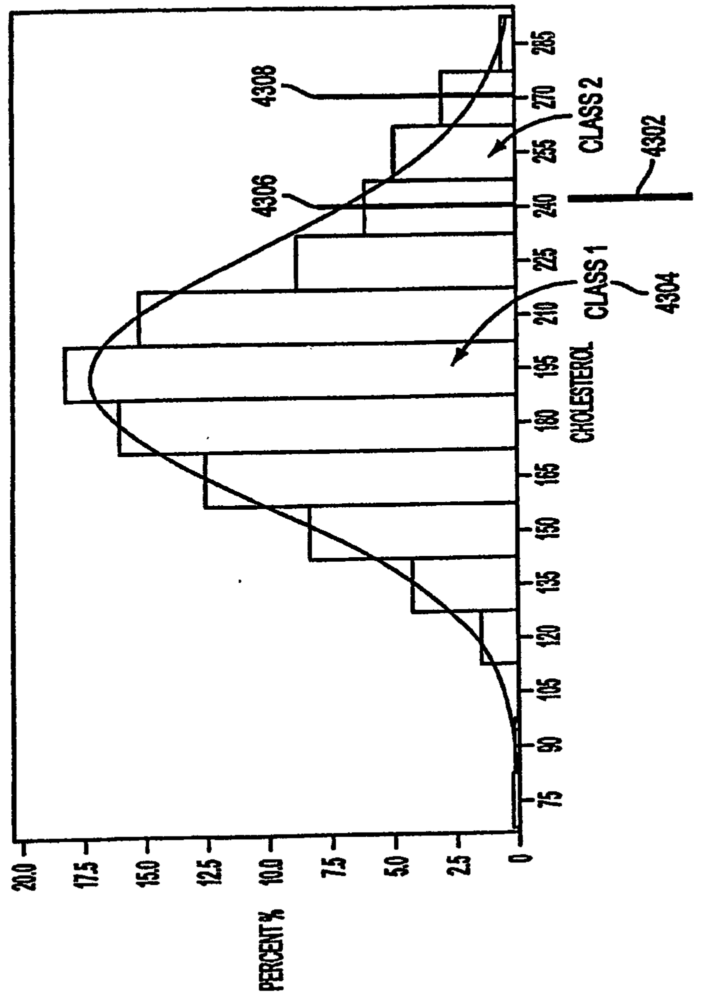

PERCENT %
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0
75
90
105
120
135
150
165
180
195
210
225
240
255
270
285
CHOLESTEROL
CLASS 1
4304
CLASS 2
4302
4306
4308

FIG. 43

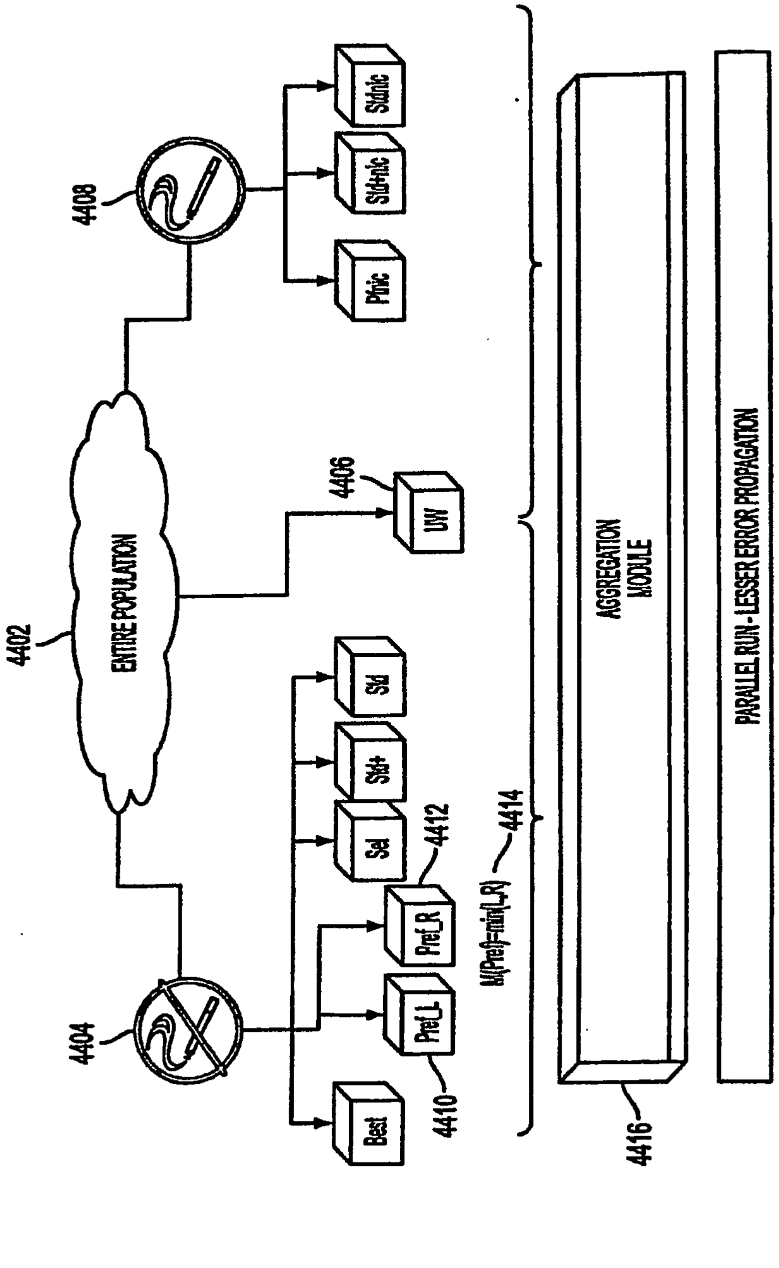
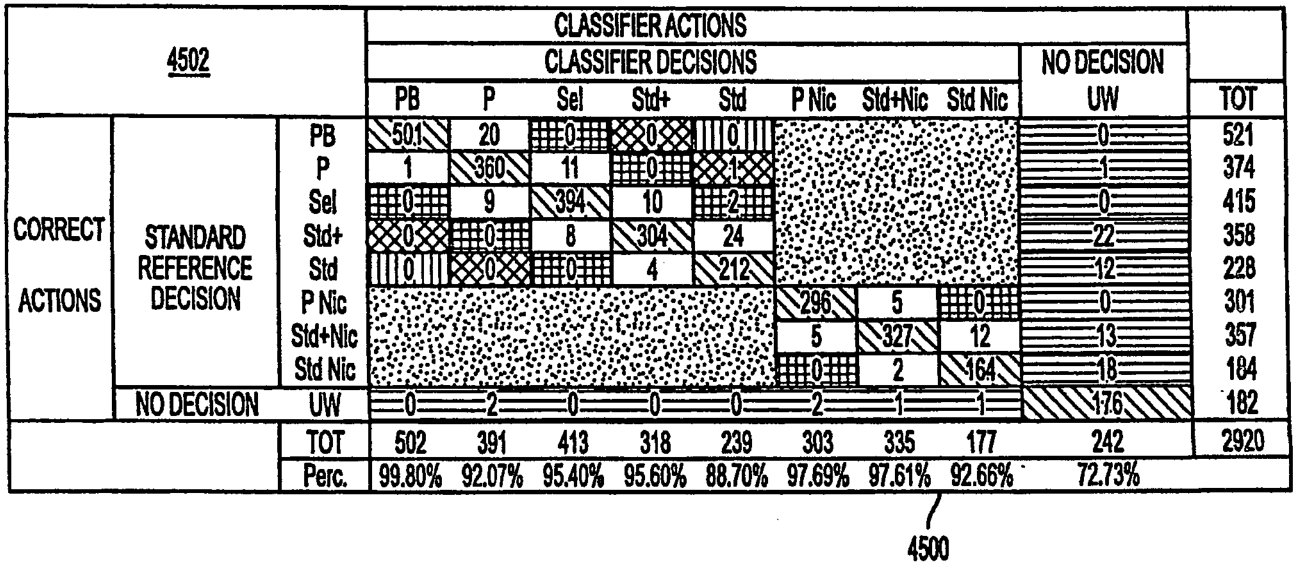

4402
ENTIRE POPULATION
4404
4406
UW
4408
Best
Pref_L
4410
Pref_R
4412
Sel
Std+
Std
Pfnic
Std+nic
Stdnic
M(Pref)=min(L,R)
4414
AGGREGATION MODULE
4416
PARALLEL RUN - LESSER ERROR PROPAGATION

FIG. 44



4502			CLASSIFIER ACTIONS									
			CLASSIFIER DECISIONS								NO DECISION	
			PB	P	Sel	Std+	Std	P Nic	Std+Nic	Std Nic	UW	TOT
CORRECT ACTIONS	STANDARD REFERENCE DECISION	PB	501	20	0	0	0				0	521
		P	1	360	11	0	1				1	374
		Sel	0	9	394	10	2				0	415
		Std+	0	0	8	304	24				22	358
		Std	0	0	0	4	212				12	228
		P Nic						296	5	0	0	301
		Std+Nic						5	327	12	13	357
		Std Nic						0	2	164	18	184
	NO DECISION	UW	0	2	0	0	0	2	1	1	176	182
		TOT	502	391	413	318	239	303	335	177	242	2920
		Perc.	99.80%	92.07%	95.40%	95.60%	88.70%	97.69%	97.61%	92.66%	72.73%	

FIG. 45

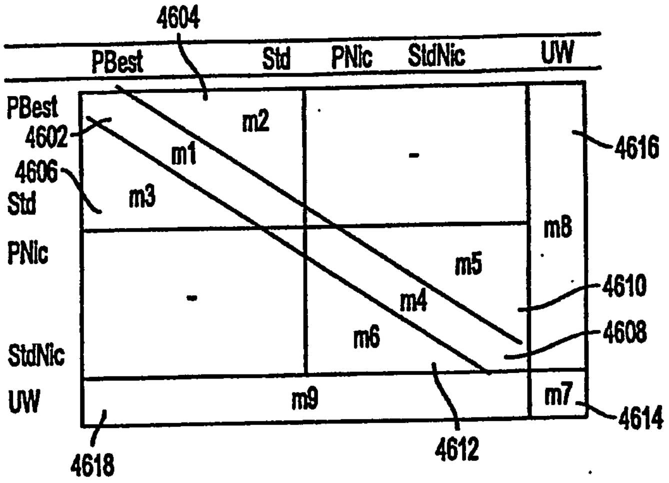
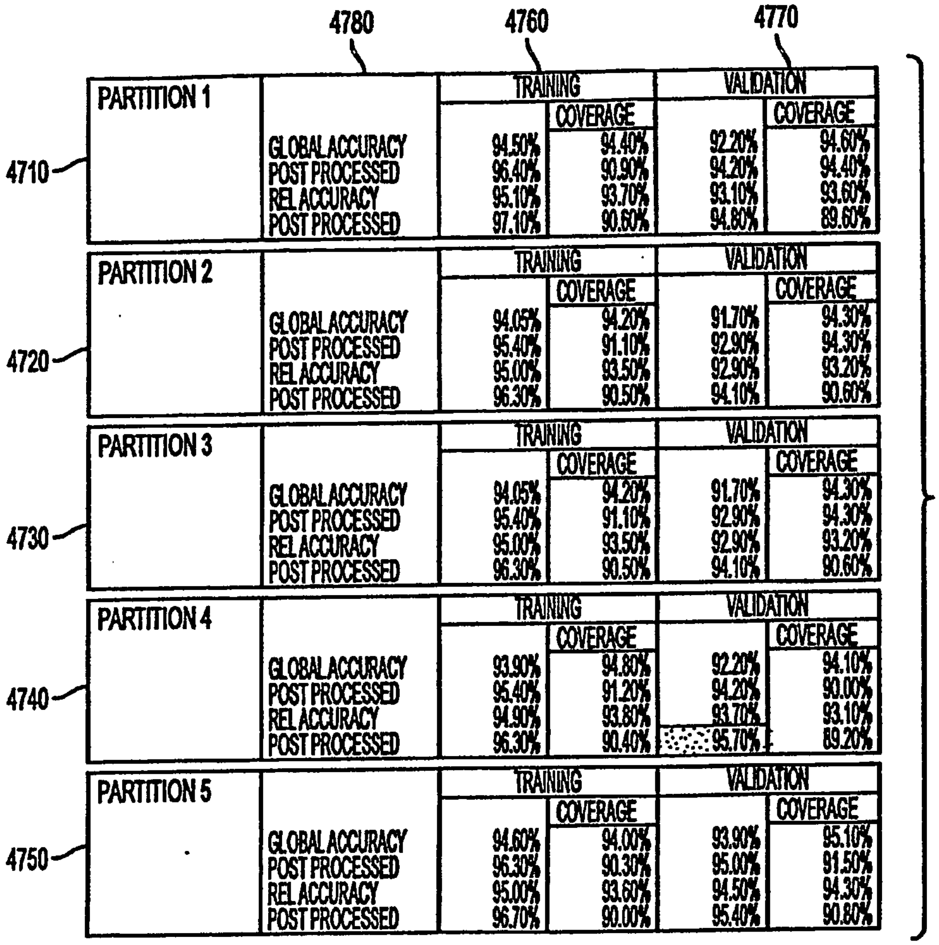

4604
PBest
Std
PNlc
StdNlc
UW
PBest
4602
4606
Std
PNlc
StdNlc
UW
m1
m2
m3
m4
m5
m6
m7
m8
m9
-
-
4616
4610
4608
4614
4612
4618

FIG. 46


4780
4760
4770
4710
4720
4730
4740
4750

PARTITION 1		TRAINING		VALIDATION	
			COVERAGE		COVERAGE
	GLOBAL ACCURACY	94.50%	94.40%	92.20%	94.60%
	POST PROCESSED	96.40%	90.90%	94.20%	94.40%
	REL ACCURACY	95.10%	93.70%	93.10%	93.60%
	POST PROCESSED	97.10%	90.60%	94.80%	89.60%

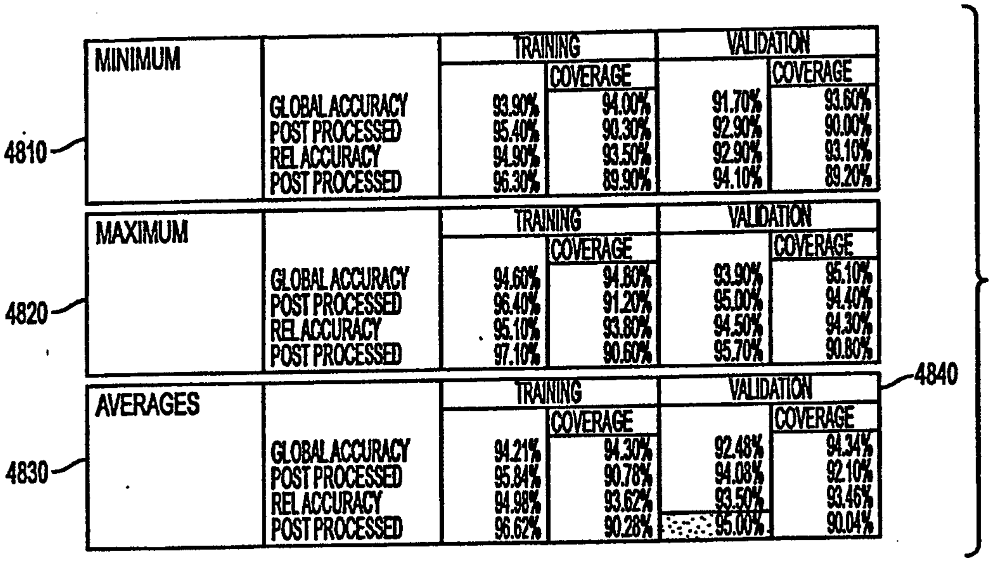
PARTITION 2		TRAINING		VALIDATION	
			COVERAGE		COVERAGE
	GLOBAL ACCURACY	94.05%	94.20%	91.70%	94.30%
	POST PROCESSED	95.40%	91.10%	92.90%	94.30%
	REL ACCURACY	95.00%	93.50%	92.90%	93.20%
	POST PROCESSED	96.30%	90.50%	94.10%	90.60%

PARTITION 3		TRAINING		VALIDATION	
			COVERAGE		COVERAGE
	GLOBAL ACCURACY	94.05%	94.20%	91.70%	94.30%
	POST PROCESSED	95.40%	91.10%	92.90%	94.30%
	REL ACCURACY	95.00%	93.50%	92.90%	93.20%
	POST PROCESSED	96.30%	90.50%	94.10%	90.60%

PARTITION 4		TRAINING		VALIDATION	
			COVERAGE		COVERAGE
	GLOBAL ACCURACY	93.90%	94.80%	92.20%	94.10%
	POST PROCESSED	95.40%	91.20%	94.20%	90.00%
	REL ACCURACY	94.90%	93.80%	93.70%	93.10%
	POST PROCESSED	96.30%	90.40%	95.70%	89.20%

PARTITION 5		TRAINING		VALIDATION	
			COVERAGE		COVERAGE
	GLOBAL ACCURACY	94.60%	94.00%	93.90%	95.10%
	POST PROCESSED	96.30%	90.30%	95.00%	91.50%
	REL ACCURACY	95.00%	93.60%	94.50%	94.30%
	POST PROCESSED	96.70%	90.00%	95.40%	90.80%

FIG. 47



MINIMUM		TRAINING		VALIDATION	
			COVERAGE		COVERAGE
	GLOBAL ACCURACY	93.90%	94.00%	91.70%	93.60%
	POST PROCESSED	95.40%	90.30%	92.90%	90.00%
	REL ACCURACY	94.90%	93.50%	92.90%	93.10%
	POST PROCESSED	96.30%	89.90%	94.10%	89.20%

MAXIMUM		TRAINING		VALIDATION	
			COVERAGE		COVERAGE
	GLOBAL ACCURACY	94.60%	94.80%	93.90%	95.10%
	POST PROCESSED	96.40%	91.20%	95.00%	94.40%
	REL ACCURACY	95.10%	93.80%	94.50%	94.30%
	POST PROCESSED	97.10%	90.60%	95.70%	90.80%

AVERAGES		TRAINING		VALIDATION	
			COVERAGE		COVERAGE
	GLOBAL ACCURACY	94.21%	94.30%	92.48%	94.34%
	POST PROCESSED	95.84%	90.78%	94.08%	92.10%
	REL ACCURACY	94.98%	93.62%	93.50%	93.46%
	POST PROCESSED	96.62%	90.28%	95.00%	90.04%

FIG. 48

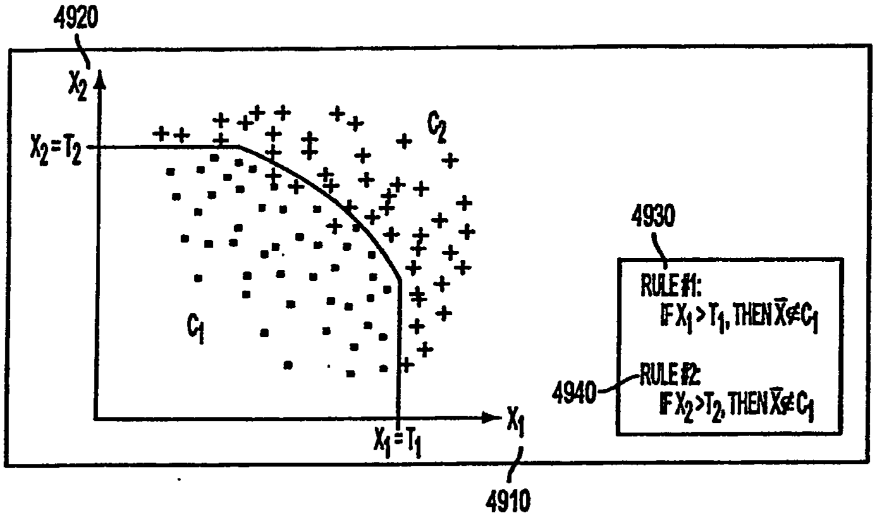

4920
X_2
$X_2=T_2$
C_2
C_1
X_1
$X_1=T_1$
4910
4930
RULE #1:
IF $X_1>T_1$, THEN $\bar{X}\notin C_1$
4940
RULE #2:
IF $X_2>T_2$, THEN $\bar{X}\notin C_1$

FIG. 49

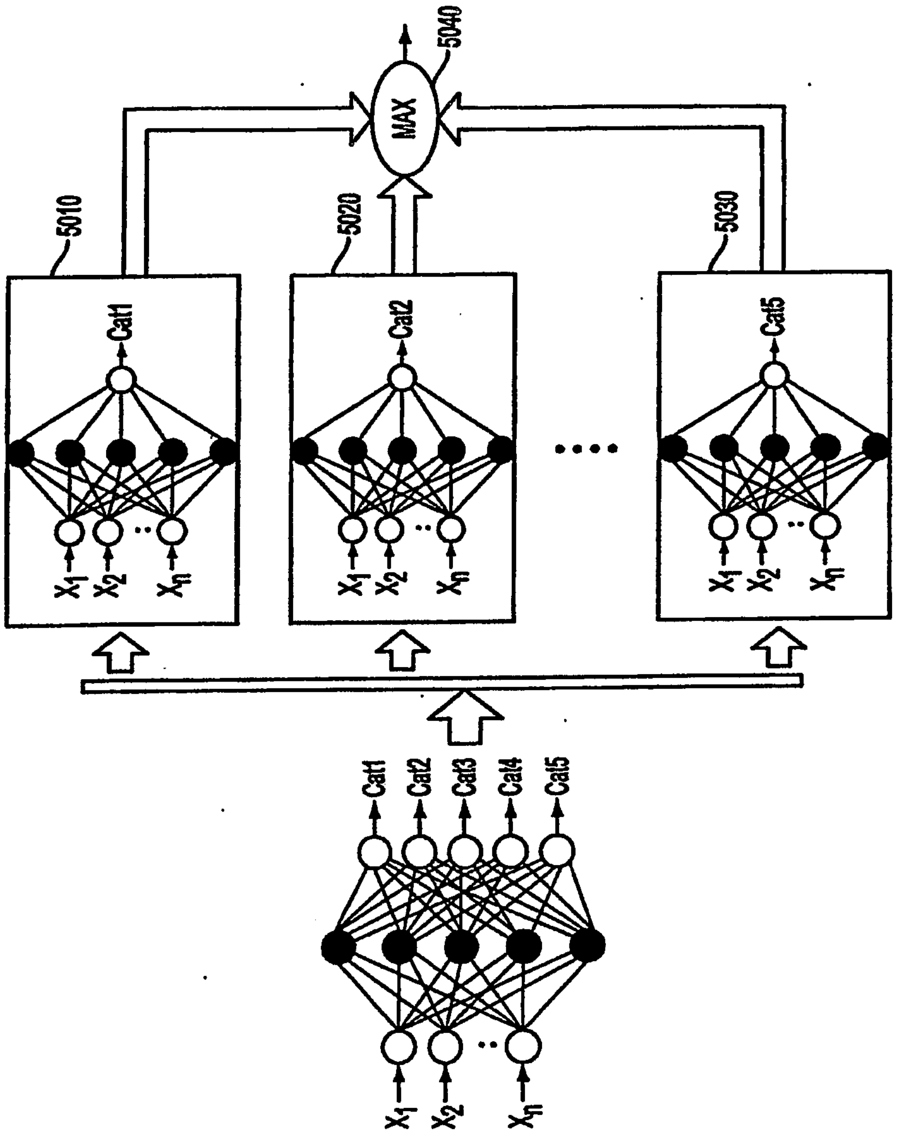

5040
MAX
5010
5020
5030
Cat1
Cat2
Cat5
X_1
X_2
X_n
Cat1
Cat2
Cat3
Cat4
Cat5

FIG. 50

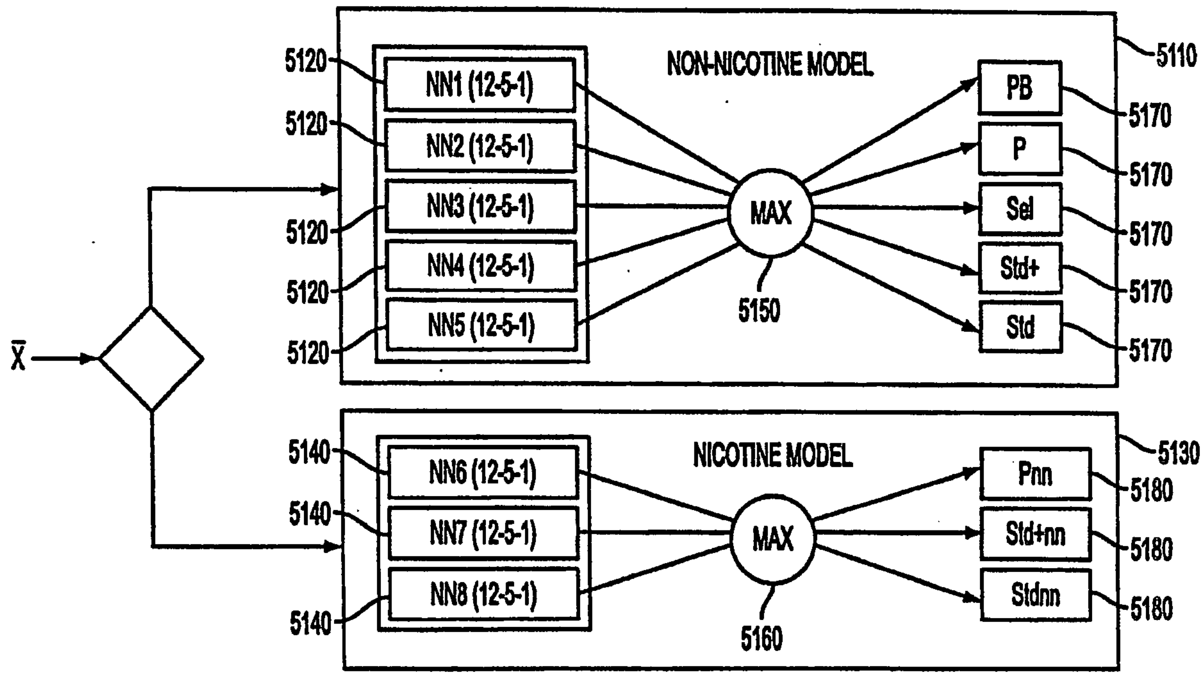

X̄
NON-NICOTINE MODEL
5110
5120
5120
5120
5120
5120
NN1 (12-5-1)
NN2 (12-5-1)
NN3 (12-5-1)
NN4 (12-5-1)
NN5 (12-5-1)
MAX
5150
PB
P
Sel
Std+
Std
5170
5170
5170
5170
5170
NICOTINE MODEL
5130
5140
5140
5140
NN6 (12-5-1)
NN7 (12-5-1)
NN8 (12-5-1)
MAX
5160
Pnn
Std+nn
Stdnn
5180
5180
5180

FIG. 51

		NN DECISION									
		PB	P	Sel	Std+	Std	P Nic	Std+Nic	StdNic	ACCURACY	
CORRECT DECISION	PB	508	4	1	0	0				99.03	513
	P	16	328	24	2	0				88.65	370
	Sel	2	12	377	19	0				91.95	410
	Std+	1	3	9	320	22				90.14	355
	Std	0	0	2	13	212				93.39	227
	P Nic						289	9	0	96.98	298
	Std+Nic						8	331	13	94.03	352
	StdNic						0	13	168	92.82	181
		96.39	94.52	91.28	90.40	90.60	97.31	93.77	92.82	93.61	2706
		527	347	413	354	234	297	353	181	2706	

FIG. 52

		NN DECISION									
		PB	P	Sel	Std+	Std	P Nic	Std+Nic	StdNic	ACCURACY	
CORRECT DECISION	PB	507	3	1	0	0				99.22	511
	P	16	326	23	2	0				88.83	367
	Sel	1	10	374	18	0				92.80	403
	Std+	1	2	6	312	21				91.23	342
	Std	0	0	2	9	211				95.05	222
	P Nic						285	9	0	96.94	294
	Std+Nic						5	318	9	95.78	332
	StdNic						0	10	165	94.29	175
		96.57	95.60	92.12	91.50	90.95	98.28	94.36	94.83	94.41	2646
		525	341	406	341	232	290	337	174	2646	

FIG. 53

BEFORE POST-PROCESSING

TEST SETS	MODEL-1	MODEL-2	MODEL-3	MODEL-4	MODEL-5	AVERAGE
COVERAGE	93.92%	93.92%	93.92%	93.92%	94.26%	93.99%
REL. ACC.	93.16%	92.98%	93.90%	94.45%	93.54%	93.61%
GLOBAL. ACC.	87.50%	87.33%	88.19%	88.72%	88.17%	87.98%

FIG. 54

AFTER POST-PROCESSING

TEST SETS	MODEL-1	MODEL-2	MODEL-3	MODEL-4	MODEL-5	AVERAGE
COVERAGE	92.01%	91.32%	91.84%	91.67%	92.70%	91.91%
REL. ACC.	93.96%	93.73%	94.52%	95.45%	94.37%	94.41%
GLOBAL. ACC.	86.46%	85.59%	86.81%	87.50%	87.48%	86.77%

FIG. 55

INTERNATIONAL SEARCH REPORT

International Application No
PCT/US2004/008626

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 G06F17/60

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5 873 066 A (FLYNN JERRY ET AL) 16 February 1999 (1999-02-16)	1,11,31
Y		2-10, 12-30, 32-40
	abstract column 2, line 9 - line 52 claim 1 figure 1 -----	
Y	WO 01/93193 A (LEXIS NEXIS ; COLLIAS SPIRO G (US); LU X ALLAN (US); MORELOCK JOHN T () 6 December 2001 (2001-12-06) abstract page 7, line 1 - page 9, line 12 ----- -/--	2-10, 12-30, 32-40

☒ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search

19 August 2004

Date of mailing of the international search report

03/09/2004

Name and mailing address of the ISA

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
Fax: (+31-70) 340-3016

Authorized officer

Falierou, C

INTERNATIONAL SEARCH REPORT

International Application No
PCT/US2004/008626

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 5 970 464 A (PEDNAULT EDWIN PETER DAWSON ET AL) 19 October 1999 (1999-10-19) abstract column 1, line 44 - column 2, line 16 -----	1-40
A	US 5 809 478 A (BABIJ K ET AL) 15 September 1998 (1998-09-15) abstract -----	1-40

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No
PCT/US2004/008626

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5873066 A	16-02-1999	NONE	
WO 0193193 A	06-12-2001	US 6684202 B1	27-01-2004
		AU 6662701 A	11-12-2001
		CA 2410881 A1	06-12-2001
		EP 1305771 A1	02-05-2003
		JP 2003535407 T	25-11-2003
		WO 0193193 A1	06-12-2001
US 5970464 A	19-10-1999	NONE	
US 5809478 A	15-09-1998	US 6584467 B1	24-06-2003